(1d-)

Special Report 83-32

November 1983

US Army Corps of Engineers
Cold Regions Research & Engineering Laboratory

A multivariable regression algorithm

George L. Blaisdell and Todd Carpenter

Prepared for
OFFICE OF THE CHIEF OF ENGINEERS
Approved for public release; distribution unlimited

Unclassified

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER Special Report 83-32	2. GOVT ACCESSION NO. AD-A136 630	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) A MULTIVARIABLE REGRESSION ALGORITHM		5. TYPE OF REPORT & PERIOD COVERED 6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s) George L. Blaisdell and Todd Carpenter		8. CONTRACT OR GRANT NUMBER(s)
9. PERFORMING ORGANIZATION NAME AND ADDRESS U.S. Army Cold Regions Research and Engineering Laboratory Hanover, New Hampshire 03755		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS DA Proj. 4A762730AT42, Tech. Area A, Work Unit 9
11. CONTROLLING OFFICE NAME AND ADDRESS Office of the Chief of Engineers Washington, D.C. 20314		12. REPORT DATE November 1983 13. NUMBER OF PAGES 44
14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office)		15. SECURITY CLASS. (of this report) Unclassified 15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release; distribution unlimited.

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

BASIC
Computer program documentation
Data analysis
Data processing
Mathematical analysis
Regression analysis

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

A BASIC algorithm has been developed that is capable of fitting a user-defined regression equation to a set of data. This best-fit-curve algorithm is unique in that it allows multiple variables and multiple forms (exponential, trigonometric, logarithmic, etc.) to be present in a single regression equation. The least-squares regression performed determines the constants for each of the regression equation terms to provide a best-fit curve. Other programs within the algorithm set allow for data entry, editing and print-out, and plotting of the raw data and their best-fit regression curve.

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE

Unclassified

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

PREFACE

This report was prepared by George L. Blaisdell, Research Civil Engineer, of the Applied Research Branch, Experimental Engineering Division, U.S. Army Cold Regions Research and Engineering Laboratory and Todd Carpenter, cooperative education student of the Mechanical Engineering Department, Michigan Technological University. Funding was provided by DA Project 4A762730AT42, Design, Construction, and Operations Technology for Cold Regions; Technical Area A, Combat Operations Support; Work Unit 9, Winter Battlefield Mobility.

The authors recognize Mary Albert and Dr. Charles Daly for their careful technical review of this report and for their helpful suggestions.

CONTENTS

ILLUSTRATIONS

TABLES

CONVERSION FACTORS: U.S. CUSTOMARY TO METRIC (SI) UNITS OF MEASUREMENT

These conversion factors include all the significant digits given in the conversion tables in the ASTM Metric Practice Guide (E 380), which has been approved for use by the Department of Defense. Converted values should be rounded to have the same precision as the original (see E 380).

Multiply	By	To obtain
inches	25.4	millimetres
pounds (force)	4.448222	Newtons
pounds (mass) per cubic foot	16.01846	kilograms per cubic metre
Btu in./hr ft^2 °F	0.1442229	W/m K

A MULTIVARIABLE REGRESSION ALGORITHM

by

George L. Blaisdell and Todd Carpenter

INTRODUCTION

It is commonly the case in data analysis that a set of experimental data is best represented by an equation. The equation allows for convenient interpolation between measured data points and provides a simple way of referring to the data. Additionally, some mathematical manipulation of data requires an equation form.

Many algorithms exist for fitting a smooth curve to a given set of data. It is generally true, however, that these algorithms are limited to a single type of mathematic form. For instance, curve fitting routines for polynomial, exponential and logarithmic functions are abundant. These curve types may produce an adequate fit over a portion or perhaps all of the experimental data. If no one type of mathematical relationship adequately fits the data, one can resort to using the cubic-spline method for interpolating data values, approximating the area integral over a specific interval or determining the first derivative at a given point.

Since some, if not most, experimental data do not strictly fit the standard mathematical relationships for which curve fitting routines exist, in the past they were either roughly approximated or left as discrete data. To overcome this difficulty, we have developed an algorithm to allow curve fitting with a user-defined mathematical expression. This expression may contain logarithmic, exponential, trigonometric and other mathematical forms singly or combined. It is also able to handle up to 20 separate independent variables. Using the least squares method, the algorithm determines the coefficients for each of the user-defined terms to produce a best fit curve. Comparisons of the goodness-of-fit of various mathematical expressions to the experimental data can be done graphically with plots or with a fit parameter generated by the algorithm.

A discussion of the theory and concepts used to develop the algorithm and details of the computer coding are described in this report. Several application examples are also illustrated.

THEORY

The least squares method of curve fitting is used by the algorithm to generate a best fit curve. This method selects a fitted curve so as to minimize the sum of the squares of the deviations of the given data points from the curve. In the linear case (two variables--one independent, one dependent) the form of curve to be fit is

$$y = a\,x + b \tag{1}$$

where

a and b = constants to be solved for

x = independent variable

y = dependent variable.

Given a set of data points, denoted by $\hat{y}_i$ (where i ranges from one to n, the number of given data points), the deviation D_i between the calculated and given values can be expressed

$$D_i = \hat{y}_i - y_i \tag{2}$$

where

$$y_i = a\,x_i + b$$

and x_- is the value of the independent variable at the given data point $\hat{y}_i$.

Continuing with the least squares method for the linear case, and using eq 2, we are given the squared deviations by

$$D_i^2 = [\hat{y}_i - (a\,x_i + b)]^2 \quad . \tag{4a}$$

The sum of the squared deviations, S, is then defined as

$$S = \sum_{i=1}^{n} D_i^2 \quad . \tag{4b}$$

A goodness-of-fit parameter can be defined from this summation as an average deviation, i.e., dividing S by n. To minimize the sum S, the partial derivatives with respect to each constant (a and b for the linear case) must be set equal to zero. This results in

$$\frac{\partial S}{\partial a} = \sum_{i=1}^{n} -2\, x_i \left[\hat{y}_i - (a\, x_i + b)\right] = 0 \tag{5a}$$

$$\frac{\partial S}{\partial b} = \sum_{i=1}^{n} -2 \left[\hat{y}_i - (a\, x_i + b)\right] = 0. \tag{5b}$$

By rearranging terms, eq 5a and 5b become

$$\sum_{i=1}^{n} \hat{y}_i\, x_i = a \sum_{i=1}^{n} x_i^2 + b \sum_{i=1}^{n} x_i = 0 \tag{6a}$$

$$\sum_{i=1}^{n} \hat{y}_i = a \sum_{i=1}^{n} x_i + b \sum_{i=1}^{n} 1 = 0 \quad . \tag{6b}$$

Equations 6a and 6b represent a series of simultaneous equations that can be solved for a and b. These constants can then be put into the equation for the general form of the curve (eq 1) to yield a best fit curve for the given $\hat{y}_i$ values.

To expand the least squares method to include more than just the fit to a straight line, a general multivariable, multiform expression is defined as

$$g = a_1\, T_1 + a_2\, T_2 + \dots + a_N\, T_N \tag{7}$$

where

g = dependent variable

$a_1 \dots a_N$ = constants to be solved for

$T_1 \dots T_N$ = relational expressions of the independent variables

N = number of relational expressions.

Using eq 7 to determine the deviation to be summed and minimized yields

$$D_i^2 = \left[\hat{y}_i - (a_1\, T_{1,i} + a_2\, T_{2,i} + \dots + a_N\, T_{N,i})\right]^2$$

$$= \left[\hat{y}_i - \sum_{j=1}^{N} a_j\, T_{j,i}\right]^2 \tag{8a}$$

where $T_{j,i}$ is the jth relational expression of the general equation (eq 7) evaluated at the ith set of known values of the independent variables. The sum to be minimized can then be defined as

$$S = \sum_{i=1}^{n} \left[\hat{y}_i - \sum_{j=1}^{N} a_j T_{j,i}\right]^2 \quad . \tag{8b}$$

Taking partial derivatives with respect to constants a_k gives

$$\frac{\partial S}{\partial a_k} = -2 \sum_{i=1}^{n} T_{k,i} \left[\hat{y}_i - \sum_{j=1}^{N} a_j T_{j,i}\right] \tag{9}$$

where k varies between 1 and N. Minimizing eq 8b by setting eq 9 equal to zero and simplifying gives

$$\sum_{i=1}^{n} T_{k,i} \left[\hat{y}_i - \sum_{j=1}^{N} a_j T_{j,i}\right] = 0 \tag{10}$$

where k = 1,2,3 ...N. Equation 9 represents the general form of the partial derivative with respect to one of the constants in the regression equation (eq 7). To obtain the set of simultaneous equations for solving the constants a_k, eq 10 is written specifically for k equal to 1 to N. This gives N equations and N unknowns and can be solved using the Gaussian elimination method with backward substitution and maximal column pivoting. Before applying this method, however, some algebraic simplification is advantageous. We can define a constant C by

$$C_k \equiv \sum_{i=1}^{n} T_{k,i} \hat{y}_i \quad . \tag{11}$$

Then using eq 10, we can further define C_k by

$$C_k = \sum_{j=1}^{N} a_j \sum_{i=1}^{n} T_{k,i} T_{j,i} \equiv \sum_{j=1}^{N} a_j A_{k,j} \tag{12}$$

which produces a series of simultaneous equations.

Although laborious, the Gaussian elimination method with maximal column pivoting is reliable and can be easily programmed, and it can be easily evaluated by a computer. Coding for the least squares method is contained in program REGRES (Appendix A).

PROGRAMMING

The multiform, multivariable algorithm was converted to programming code for an HP 9845B minicomputer (user manuals referenced in Appendix B). The algorithm is readily adaptable to any computing system with an interpreter; however, graphics and matrix manipulation capabilities should be available.

For convenience, the algorithm has been broken into five parts that are each separately programmed (Appendix A). The first program, AUTOST, provides for data input (either from keyboard entry or reading from a data file already generated by this set of programs) and access to the other programs (REGRES, PLOTS, EDIT, LIST). Program REGRES performs the actual regression on the user-defined equation. PLOTS provides a graphical output for the data points and a curve or series of curves of the calculated equation. The EDIT program is a data editor that allows additions, deletions or changes in the input data. Program LIST provides a hard copy listing of the input data.

Up to 20 subfiles can be defined at the time of data input. It is important that the data be organized in these subfiles in a logical manner (i.e. holding all variables constant except one). All operations contained in the five programs can be applied to the complete data set or any combination of subfiles desired.

<u>Autostart program</u>

AUTOST is an observation definition program that also provides access to the other programs in the set. All keyboard input of data--including subfiles, subfile names and all observations--can be stored on the assigned mass storage device in a user-defined filename that does not already exist. This data file can be called during future program runs to avoid repeated keyboard entry. Special function keys K0-K6 are defined by this program by accessing the previously defined and stored key file K1 (file K1 should always exist on the assigned mass storage device). Special function key definitions are as follows:

- K0 - stops data input into the current subfile and allows data to be put into the subsequent subfile if desired.
- K1 - calls the regression equation program (REGRES) and performs the regression.

K2 - calls the PLOTS program which plots the most recently calculated regression with the current data.

K3 - calls the EDIT program where the current data may be modified by adding, deleting or changing observations and add'·ʒ subfiles.

K4 - stores the current data under a user-defined filename (provided that filename does not already exist).

K5 - calls the LIST program which lists all the data or certain subfiles on the internal printer.

K6 - stops the program.

Key K0 is only valid during keyboard entry of data and keys K1 - K6 are only active when the prompt "select K1-K6" appears on the screen. An error message will result if the keys are pressed at any other time.

An outline of the program flow for AUTOST is as follows:

1. The user is asked if the special function key definition prompts should be suppressed. If yes, go to step 3 ("Y" or "YES", CONT); if no, go to step 2.

2. The following list is output on the internal printer:

K0 = Stop data input for current subfile
K1 = Regression
K2 = Plots
K3 = Edit
K4 = Store
K5 = List
K6 = Quit.

3. The program asks if stored data are being used. If yes, go to step 11; if no, go to step 4.

4. Enter the number of variables for the data that are to be input from the keyboard.

5. Enter the name of the subfile that the data are to be stored under (must be no more than five characters long).

6. Input the data for all variables in the current observation. If this is the first observation of the subfile, all variables must be included and separated by commas. After data entry for the first observation is complete, an echo of the data will be printed on the screen. Following entry of the first observation, the user may indicate repeated

values of any variable with a "+" sign (all variables must still be separated by commas). It is important that all variables be satisfied for each observation, whether it is a repeated value or zero. Dependent and independent variables are not differentiated at this point but will be identified later in REGRES.

7. If all data entry for the current subfile is complete, press K0 and go to step 8; if not, repeat step 6.

8. The program requests a new subfile. If more subfiles need to be entered (answer "yes" to the prompt), go to step 5; if no, go to step 9.

9. If the data are <u>not</u> to be stored, answer "no" to the "data stored?" prompt and go to step 12. If the data are to be stored, answer "yes" and go to step 10.

10. The program requests a filename (one that does not already exist) and then stores the data. Go to step 12.

11. Enter the filename under which the data were previously stored by this program.

12. Select key K1-K6 to access the other programs.

Regression program

The REGRES program performs a least-squares regression on the data defined by AUTOST with a multiform, multivariable equation of up to 20 terms that is supplied by the user. Each term is input as a character string (up to 70 characters long) and is defined by the matrix notation X(A,n), where A indicates the observation number and n denotes the number of the independent or dependent variables. For example, a regression on the data in Table 1 may be desired using the form

$$y = a \ln(x) + bx^3 + cx^2z + d \qquad (13)$$

where

y = dependent variable

x and z = independent variables

$\ln(x)$ = first regression term

x^3 = second regression term

x^2z = third regression term

and a,b,c and d are regression constants and the fourth regression term is 1.

Under the format required by REGRES, eq 13 would appear as

$$x(A,2) = a * \mathrm{LOG}\ (x(A,1)) + b * x(A,1) ** 3 + c * x(A,1) ** 2 * x(A,3) + d \qquad (14)$$

where the variables x, y and z are being denoted by the numbers 1, 2 and 3, respectively, in the matrix notation.

To ease the input of laborious equations, an option to define intermediate constants for the regression equation was made available. Such constants, called user-defined terms, are input as a complete program line so it is important that proper syntax is used. User-defined terms are advantageous when the regression equation contains repeatable sections. In the example above, one could simplify the equation by creating a user-defined term

$$N1 = x(A,\ 1)$$

and restructing the regression eq (eq 13) as

$$x(A,2) = a * \mathrm{LOG}(N1) + b * N1 ** 3 + c * N1 ** 2 * x(A,3) + d. \qquad (15)$$

Clearly, this example does not show much improvement, but user-defined terms can also be used with any programmable statements.

The user is restricted in the variable names that can be used to avoid interfering with existing variables. To be safe, N1, N2, N3, etc., are recommended variable names.

The program flow for REGRES can be summarized as follows (characters in brackets indicate correct responses for the example above):

1. If this is not the first time REGRES has been run since power was turned on, go to step 9; otherwise, go to step 2.

2. The program asks if the user wants to define any user-defined terms. If not, go to step 5 [N]; otherwise, go to 3.

3. Enter the number of user-defined terms.

4. Enter each user-defined term as a complete expression within double quotes.

5. Enter the number of terms in this regression [4].

6. Enter the numerical index of the dependent variable [2].

7. Enter, one term at a time, the terms of the regression equation within double quotes:

["LOG (X(A,1)"] (then press continue)

[" X(A,1) ** 3"]

[" X(A,1) ** 2 * X(A,3)"]

[1] or ["X(A,1) ** 0"].

8. If a program or data file named BUFFER already exists on the assigned mass storage medium, its name must be changed or it will be deleted by the REGRES program. Go to step 16.

9. Since REGRES has been run before, the program still has the previous equation defined. It now asks if the previous equation is to be used again. If yes, go to step 18; if no, go to step 10.

10. The program asks if any of the previous user-defined terms are to be retained. If not go to step 13; otherwise, go to step 11.

11. Enter the number of user-defined terms to be retained and their numerical indicies.

12. Enter any additional user-defined terms.

13. The program asks if any of the terms of the previous regression equation are to be retained. If not, go to step 15; if yes, go to step 14.

14. Enter the number of terms to be retained and their numerical indicies.

15. Enter the terms necessary to complete the regression equation, as in step 7.

16. The program now asks if all the subfiles are to be used in the regression. If yes, go to step 18; if not, go to step 17.

17. Enter the number of subfiles and the subfile numbers wanted for the regression.

18. Following display of the equation and the solved constants, the program asks if the user wants a plot of the data and the regression. If no, the program branches to step 12 of AUTOST; if yes, the program branches to program PLOTS.

Plots program

For aid in determining the goodness-of-fit of the regression, a plot of the experimental and calculated data can be produced with the PLOTS program. The program will plot the dependent variable defined in REGRES against any of the other variables. Two plotting options are available to attempt to clearly display the relationships of several variables on one

Table 1. Example data for multivariable regression (values are for variable y [n=2]).

x (n=1)	z(n=3) 1	2	3	4	5	6	7	8	9	10
1	58	62	65	69	73	76	80	83	37	91
2	90	104	119	133	147	162	176	191	205	219
3	157	190	222	255	287	319	352	394	417	449
4	277	334	392	449	507	565	622	680	737	795
5	463	553	643	733	823	913	1003	1093	1133	1273
6	731	860	990	1119	1249	1379	1508	1639	1767	1897
7	1096	1272	1448	1625	1801	1978	2154	2330	2507	2683
8	1573	1803	2034	2264	2494	2725	2955	3186	3416	3646
9	2177	2469	2760	3052	3343	3635	3927	4218	4510	4801
10	2924	3284	3644	4004	4364	4724	5084	5444	5804	6164

graph. To illustrate these, the data from Table 1 are used. All of the x,y combinations are shown plotted in Figure 1. The first plotting option displays the dependent variable on the ordinate with one of the independent variables on the abscissa and a line representing a constant value of a second independent variable as calculated by the regression program (Fig. 2). The second type of display plots the data points with the dependent variable on the ordinate and the chosen independent variable on the abscissa. It also plots the calculated value from the regression using all of the variables (Fig. 3).

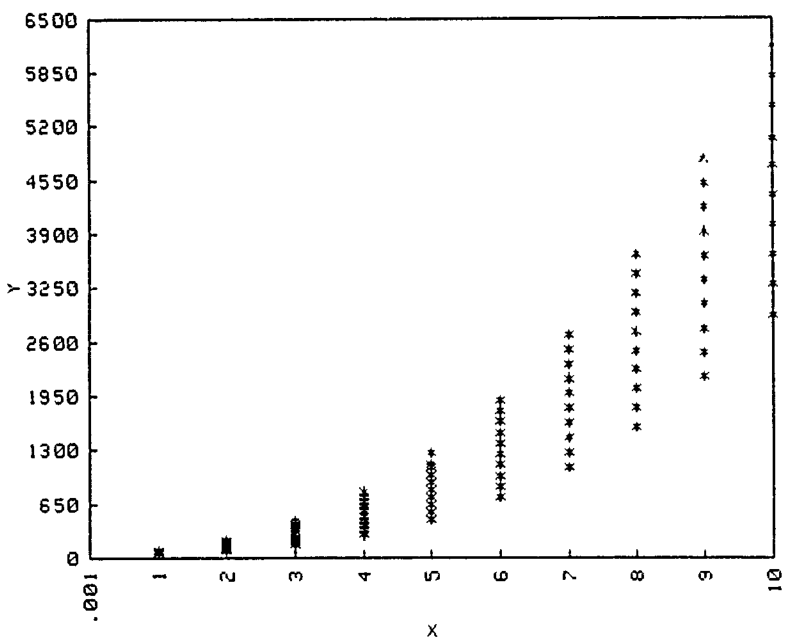


Figure 1. Data from Table 1, x and y only.

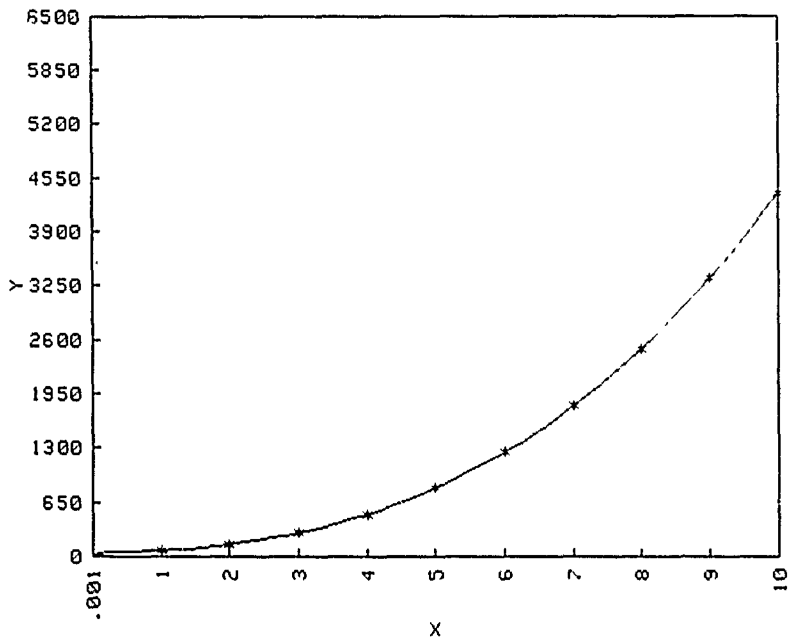


Figure 2. Program PLOTS output for example data of Table 1 (for z = 5); the ordinate displays the dependent variable (y), the abscissa displays an independent variable (x) and the plotted curve represents a constant value of a second independent variable (z = 5).

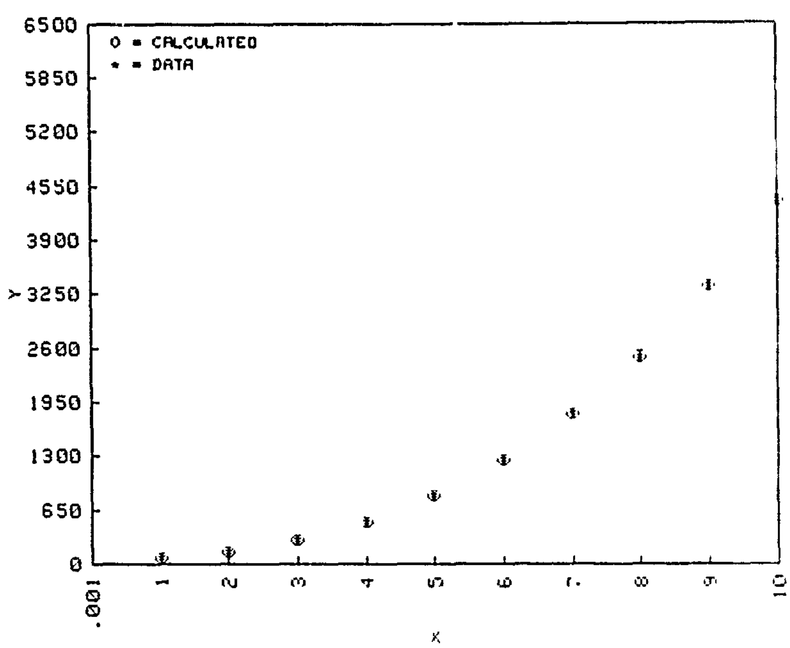


Figure 3. Program PLOTS output for example data of Table 1 (for z = 5); the ordinate displays the dependent variable (y), the abscissa displays an independent variable (x); the *'s represent actual data points and the o's plot calculated points (from the regression) using all of the independent variables.

The flow of program PLOTS follows:

1. Define the independent variable for the abscissa with its index number.

2. If all subfiles are to be plotted, go to step 4.

3. Enter the number of subfiles and the specific subfile numbers that are to be used in plotting.

4. The program displays the maximum and minimum values of the variable chosen for the plot axes. The user is asked to enter the limits for the axes' scaling.

5. Enter the label for the abscissa (independent variable) and the ordinate (dependent variable).

6. To plot, holding all variables constant except one (the independent variable chosen in step 1), go to step 11.

7. The program plots each data point (*) and the calculated points from the regression program (0) for each data point using all variables.

8. The program asks if a hard copy is desired, if not it branches to step 10.

9. A hard copy is printed with the regression equation and a goodness-of-fit parameter (a fit parameter of 0 is a perfect fit). If a different plot is wanted, go to step 1.

10. If another regression is desired, the program branches to step 1 of program REGRES. If no additional regressions are desired, the program branches to step 12 of AUTOST.

11. The program requests entry of a constant value for all variables except the one identified in step 1.

12. The plot is displayed. If another curve is desired, go to step 13; if not, go to step 8.

13. The program requests the number of variables to be changed, which ones (identified by their numerical index) and their new values. Go to step 12.

Edit and list programs

Program EDIT allows the user to change the values of observations, add and delete observations, and add complete subfiles to the existing files. All editing features include either subfile or absolute addressing. Absolute addressing numbers all observations sequentially from the first

observation of the first subfile to the last observation of the last subfile. Subfile addressing renumbers all observations in each subfile such that the first observation of each subfile is referenced as the first observation. The EDIT program offers menu-type edit feature selection. The delete feature first lists the values of all variables of the observation referenced and offers an escape by asking if it should be deleted.

The LIST program provides a hard copy listing of some or all subfiles. It prints the subfile name at the top of each subfile and references all observation numbers using subfile addressing.

ALGORITHM APPLICATION

In the study of snow mechanics, one of the common techniques used to classify snow compressibility is the plate-sinkage test. A circular or rectangular plate driven by a hydraulic or manual ram compacts a volume of snow, changing its original density to critical density (approximately 3.12×10^{-5} lb/ft^3). At the critical density, the volume of snow is reduced very little or not at all. Increased pressure from the plate results in constant-volume flow or movement of the snow away from the plate.

Measurements taken during a plate-sinkage test include the force on the plate and an indication of the amount of sinkage or vertical compaction experienced between the beginning of the test (zero force) and each force reading. From the force-sinkage data, the compaction energy can be determined by integration between the zero-sinkage and critical sinkage (sinkage at which critical density is reached) limits. This integration can be performed numerically to yield acceptable results. Integration of the curve or comparison of the nature of the compaction relationship between various snows, however, is accomplished best with a mathematical representation. Using the multivariable algorithm, we can determine the mathematical relationship between force and sinkage.

Given the force versus sinkage data in Table 2, a best-fit mathematical expression is desired. Since we know that the curve is somewhat exponential in nature, the first regression equation form used in REGRES is

$$F = a\ e^{z} \qquad (16)$$

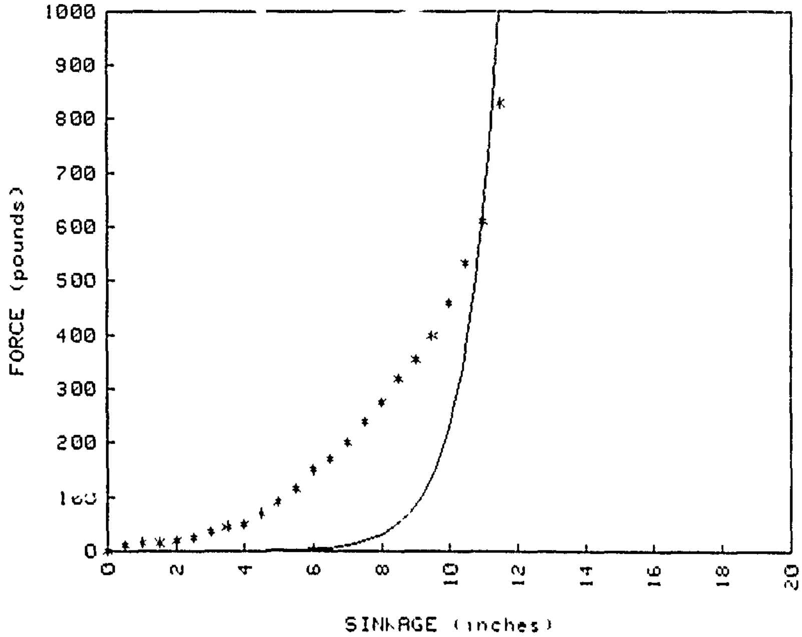
which gives $a_1 = 0.01347$ and from PLOTS shows a rather poor fit (Fig. 4). The second equation type attempted is a 4th order polynomial and it results

Table 2. Typical force versus sinkage data for a plate-sinkage test in snow.

Force (lb)	Sinkage (in.)
0	0
10	0.5
15	1.0
15	1.5
20	2.0
25	2.5
35	3.0
45	3.5
50	4.0
70	4.5
90	5.0
115	5.5
150	6.0
170	6.5
200	7.0
240	7.5
275	8.0
320	8.5
355	9.0
400	9.5
460	10.0
535	10.5
610	11.0
830	11.5

in the coefficients and fit shown in Figure 5. This is a much better approximation to the data, however, near the upper and lower ends of the data the polynomial shows an increasingly poor fit. (This is a common behavior for polynomial curve fitting.) Recognizing that the exponential example showed a horizontal curve near the low sinkage values and became steeper at the high sinkage values, we next attempt a combination of exponential and polynomial forms. The equation used by REGRES for this trial is

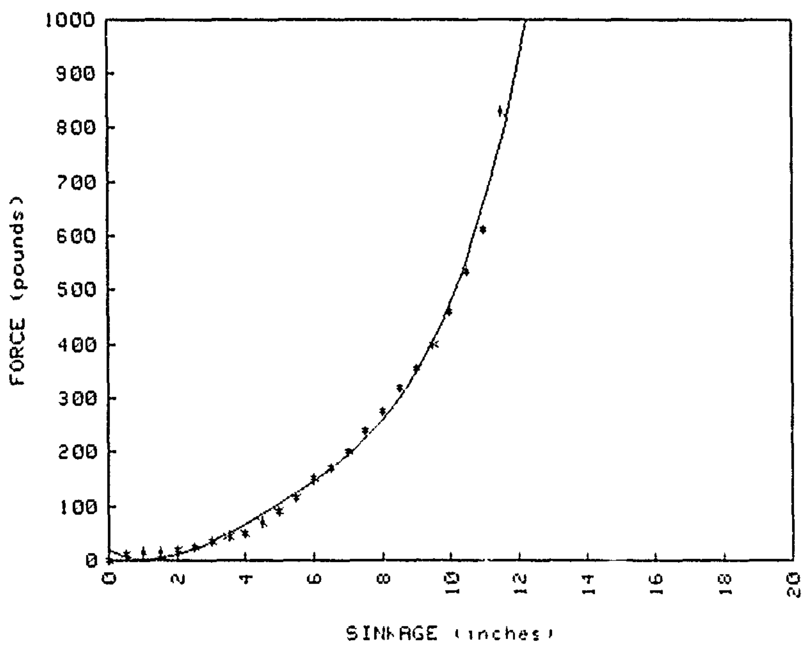
$$F = a_1 + a_2 z + a_3 z^2 + a_4 z^3 + a_5 z^4 + a_6 e^z \ . \qquad (17)$$




```
Fx=a*EXP(X(A,2))
a=1.03470980775E-02

Goodness-of fit-is   23122.703
```

Figure 4. Exponential curve fit on plate-sinkage data.

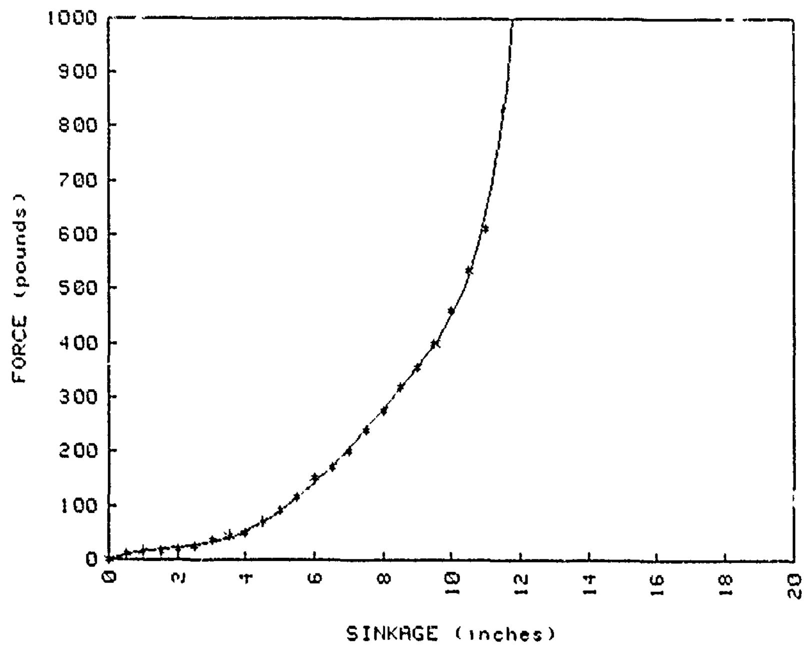



```
Fx=a*N1^4+b*N1 3+c*N1 2+d*N1+e*1
a=.148761854601
b=-2.7159984382
c=20.4106865246
d=-35.7314824877
e=19.0238579006

N1=X(A,2)

Goodness-of fit-is   356.581
```

Figure 5. Fourth-order polynomial curve fit on plate-sinkage data.




```
Fx=a*H1^4+b*H1^3+c*H1^2+d*H1+e*1+f*H2
a=-.196148723165
b=3.4161178319
c=-14.1977673419
d=28.1674845102
e=-1.63662772089
f=6.18900113498E-03

H1=X(A,2)

H2=EXP(X(A,2))

Goodness-of fit-is   49.825
```

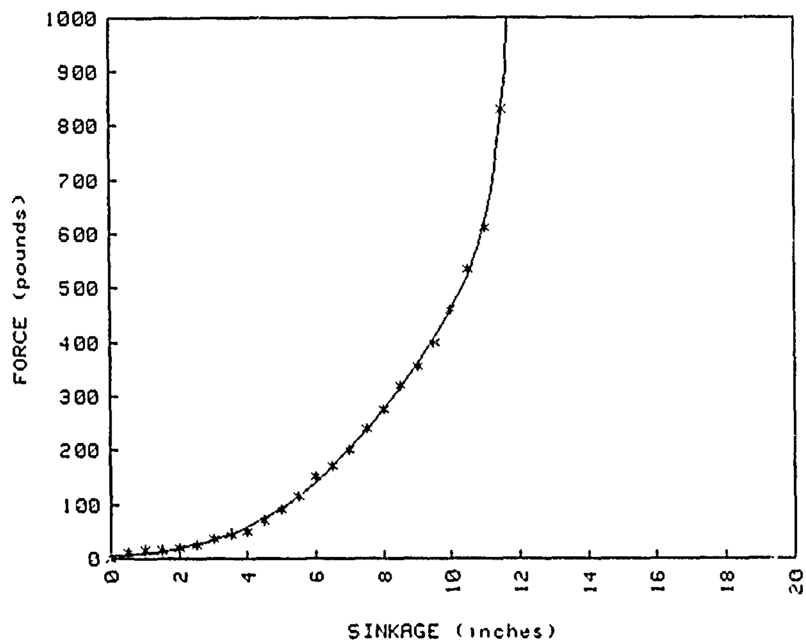
Figure 6. Fourth-order polynomial plus exponential curve fit for plate-sinkage data.

Performing the regression with eq 17 and the data in Table 2, we obtain an excellent fit (Fig. 6). Another form of combining the polynomial and exponential fits is shown in Figure 7 where

$$F = e^{z} (a_1 + a_2 z + a_3 z^2 + a_4 z^3 + a_5 z^4) + a_5 \quad . \qquad (18)$$

This also results in a good curve fit and comparison of the goodness-of-fit parameter is necessary to determine that eq 18 more closely approximates the data.

Typical heat transfer problems demonstrate the multivariable power of this regression algorithm. A case in point involves the placement of flat heat flux sensors on a curved surface, such as a pipe. Heat flux sensors (HFS) are thin wafers of a material with known thermal conductivity. Thermocouples are attached to each flat side of the wafer. The output from the HFS is a voltage that is proportional to the heat flux through the sensor. The flat, inflexible HFS is attached to the pipe by an epoxy. The epoxy, in addition to securing the sensor to the pipe, fills in the void




```
Fx=a*N2*N1^4+b*N2*N1^3+c*N2*N1^2+d*N2*N1+e*N2+f*1
a=3.57915750310E-04
b=-1.55831702843E-02
c=.2587583991
d=-1.9476292602
e=5.62542674891
f=-.797026549158

N1=X(A,2)

N2=EXP(X(A,2))

Goodness-of fit-is   20.337
```

Figure 7. Fourth-order polynomial times exponential curve fit for plate-sinkage data.

space between the flat HFS and the curved pipe. This attachment alters the factory calibration for thermal conductivity, which is for application on a flat surface. Using experimental data from an insulated pipe section with a HFS attached, we can determined a new calibration for thermal conductivity by relating the heat flux out of the pipe to the sensor voltage output. This calibration relation can easily be represented by a smooth curve. Different insulation thicknesses generate a series of curves. Additionally, with different pipe insulation thermal conductivities, pages of these series of curves are generated. Likewise, with different glues for attaching the sensor, volumes of pages of series of curves are required. It soon becomes apparent that a mathematical relation is necessary to calibrate the heat flux out of the pipe with the HFS reading, the type of glue, the type and thickness of the insulation, and the diameter of the pipe.

The first step in fitting an equation to these data is to break the data into subfiles where only one variable is changing. This isolates the

effect each variable has on the heat flux out of the pipe and the heat flux through the sensor. Starting with the basic equation for heat flux out of the pipe

$$\phi_p = \frac{K_i \ \phi_s}{\ln \dfrac{r_p + t_i}{r_p}} \tag{19}$$

where

K_i = insulation thermal conductivity

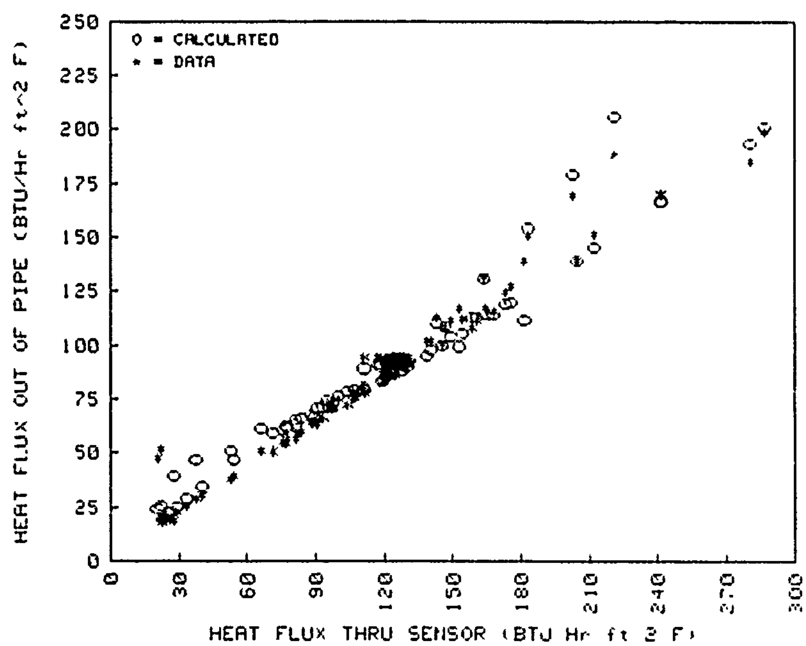
ϕ_p = heat flux out of pipe

ϕ_s = heat flux through sensor

r_p = radius of the pipe

t_i = thickness of insulation,

we determine the effect of each variable on the heat flux. All the terms




```
F(=a*X(A,2)*X(A,9)/(LOG((X(A,4) 2+X(A,5)) [illegible] A,4) 2)))+b* [illegible]
d*(X(A,1) (1/3)) X(A,9)+e*1 /(A,9)+f*X(A,9)* (A,3)+g* (A,4)* (A,5 +h* [illegible]
4))*X(A,5)+i*X(A,2)*X(A,9) (X(A,4)* (A,5))+j*1
a=4.26558807638E-02
b=.549739257301
c=2.52601427196
d=4.69904297259
e=-23[illegible] 577621555
f=3.63389190427E-0[illegible]
g=4.9489026932
h=-.630665070255
i=.290693019949
j=13.1514385954

Goodness-of fit-is   57.053
```

Figure 8. Multivariable curve fit for heat flux through an insulated pipe with a flat heat flux sensor.

of these equations can then be included in a final regression on all the subfiles. The final form of the best fit equation is

$$\phi_p = \frac{aK_i\,\phi_s}{\ln \frac{r_p + t_i}{r_p}} + b\,K_i\phi_s + cK_i + \frac{dK_e^{1/3}}{\phi_s} + \frac{e}{\phi_s}$$

$$+ f\,\phi_s\,K_s + g\,r_p\,t_i + h\,K_s\,r_p\,t_i + \frac{i\,K_i\,\phi_s}{r_p\,t_i} + j \quad (20)$$

where K_e is the epoxy thermal conductivity and K_s is the sensor thermal conductivity. The constants a-j and a plot of the calculated and raw data are shown in Figure ~. This figure appears cluttered because all the subfiles of data were plotted. This can be avoided by either expanding the scale to look at discrete sections of the data or plotting one subfile at a time.

CONCLUSION

The multivariable, multiform regression algorithm presented can be seen to be a useful alternative to single form regressions. The algorithm allows data that do not fit one of the standard equation forms (polynomial, exponential, etc.) to be mathematically characterized by a best-fit curve. This allows the data to be easily integrated and differentiated and provides an accurate interpolating equation. An equation representation of the data also provides a method of comparing the variable relationships between separate sets of data.

Specific application of the algorithm is shown for snow compaction with a plate-sinkage device. The results can be used for predicting vehicle motion resistance when the vehicle is operating in snow. The algorithm is also applied to the problem of heat flow around a pipe. Measured heat flow data can be fit with a regression equation and a comparison of actual and analytical solutions completed.

APPENDIX A: ALGORITHM CODING.

```
10    REM  --> PROGRAM AUTOST
20    ! STARTER PROGRAM FOR MULTI-VARIABLE REGRESSION.
30    PRINT PAGE,TAB(13);CHR$(129);"USER INSTRUCTIONS FOR MULTI-VARIABLE REGRESS
ION:";CHR$(128)
40    PRINT LIN(2);"This program is the starter program for a set of five progra
ms"
50    PRINT "that can perform a series regression of a user-defined equation wit
h "
60    PRINT "up to 500 observations and 20 variables per observation."
70    PRINT LIN(1);"The data is input from the terminal and can later be stored
for future use."
80    PRINT "All data is broken into subfiles that can be any length so long as
the 500 "
90    PRINT "observation maximum is not exceeded.  Up to 20 subfiles can be spec
ified,"
100   PRINT "and any regression or plotting function can be performed with any c
ombination"
110   PRINT "of these subfiles."
120   PRINT LIN(1);"Any variable can be assigned the dependent variable by desig
nating"
130   PRINT "the index number of the variable desired, therefore all variables a
re treated"
140   PRINT "the same.  Plots can be made with any variable as the independent v
ariable"
150   PRINT "except the dependent variable.  The dependent variable can not appe
ar in the"
160   PRINT "regression equation."
170   DISP "PRESS 'CONT' TO CONTINUE."
180   PAUSE
190   PRINT PAGE,TAB(30);CHR$(132);"PROGRAM OPTIONS";CHR$(128)
200   PRINT LIN(2);"The following programs and options are available:"
210   PRINT LIN(1),TAB(5);"1) Keyboard data input with subfile definition, or re
ad data from"
220   PRINT TAB(8);"a previously stored data file."
230   PRINT LIN(1),TAB(5);"2) Regression of user defined series equation such as
 Y=a*X^3+b*ln(X)+c"
240   PRINT TAB(8);"where a,b,and c are the constants that the program solves."
250   PRINT LIN(1),TAB(5);"3) Plot the regression equation with a data scatter p
lot to show correlation"
260   PRINT LIN(1),TAB(5);"4) Data editing including observation changes, additi
ons, and deletions with"
270   PRINT TAB(8);"subfile additions and data file merging."
280   PRINT LIN(1),TAB(5);"5) Store the current data in a user-defined filename.
"
290   PRINT LIN(1),TAB(5);"6) List all the data or only selected subfiles."
300   DISP "PRESS 'CONT' TO CONTINUE."
310   PAUSE
320   PRINT PAGE,TAB(22);CHR$(132);"DEFINITIONS OF USER-DEFINED KEYS";CHR$(128)
330   PRINT LIN(2),TAB(5);"The user-defined keys K0-K6 are used to quickly acces
s the various"
340   PRINT "program options.  K0 is used during data input to suspend input to
the current"
350   PRINT "subfile.  It is used only after all data has been recorded into the
 current "
360   PRINT "subfile.  The option of adding a subfile is then made available, af
ter which"
370   PRINT "all data is put into the new subfile."
```

```
380   PRINT LIN(1),TAB(5);"The keys K1-K6 access the other program options, and
should be used "
390   PRINT "only when the prompt  SELECT K1-K6  appears.  The plot option shoul
d not be"
400   PRINT "used before a regression has been performed since it does not have
an "
410   PRINT "equation to perform."
420   PRINT LIN(1),TAB(5);"The option of getting a hard copy reminder of what th
ese keys do"
430   PRINT "is now available, you can suppress the prompts if so desired."
440   OPTION BASE 1
450   DIM A$(20)[12],B$(20)[12]
460   COM X(500,20),V$(20)[160],Nsub$(20)[10],Ab(20,21),INTEGER Dep,Subf(20),Nsu
b,Nobs,Nv,Nt,M$(20)[150],Term
470   INPUT "SUPRESS KEY PROMPTS?",S$
480   GOSUB Check
490   IF Check=1 THEN 470
500   IF UPC$(S$[1,1])="Y" THEN 590
510   PRINTER IS 0
520   PRINT LIN(2),"K0 = STOP DATA INPUT FOR CURRENT SUBFILE."
530   PRINT "K1 = REGRESSION."
540   PRINT "K2 = PLOT."
550   PRINT "K3 = EDIT."
551   PRINT "K4 = STORE."
570   PRINT "K5 = LIST "
580   PRINT "K6 = QUIT"
590   PRINTER IS 16
600   GCLEAR
610   EXIT GRAPHICS
620   LOAD KEY "K1:T15"
630   PRINT PAGE
640   INPUT "ARE YOU USING STORED DATA?",S$
650   GOSUB Check
660   IF Check=1 THEN 640
670   IF UPC$(S$[1,1])="N" THEN 770
680   ON ERROR GOTO Badfile
690   INPUT "FILE NAME?",Filnm$
700   ASSIGN #1 TO Filnm$
710   OFF ERROR
720   PRINT PAGE
730   READ #1;Nv,Nobs,Nsub
740   REDIM X(Nobs,Nv),Subf(Nsub+1),Nsub$(Nsub)
750   READ #1;Nsub$(*),Subf(*),X(*)
760   GOTO 1360
770   INPUT "NUMBER OF VARIABLES? (\= 20)",Nv
771   REDIM X(500,Nv)
780   IMAGE #,2X,4A,2D
790   Nsum=0
800   IMAGE #,1X,7A
810   IMAGE #,1X,5D.4D
820   Nobs=Subf(1)=0
830   Nsub=0
840      Nsub=Nsub+1
850      DISP "NAME OF SUBFILE #";Nsub;" = ?";
860      INPUT "  (<= 5 CHARACTERS)",Nsub$(Nsub)
870      PRINT CHR$(27)&"m"
880      PRINT PAGE,TAB(30);"        SUBFILE - ";Nsub$(Nsub)
890      PRINT "OBS#"
900      FOR B=1 TO Nv
910         PRINT USING 780;"VAR#";B
920      NEXT B
930      PRINT
940      PRINT CHR$(27)&"l"
950      REDIM A$(Nv),B$(Nv)
960      Bcount=0
970         Nobs=Nobs+1
980         Bcount=Bcount+1
990         DISP "INPUT ALL VARIABLES FOR OBSERVATION #";Bcount;"     '+' = REPE
AT      ";
```

```
1000        MAT INPUT A$
1010        IF Bcount<>1 THEN 1050
1020        FOR C=1 TO Nv
1030           B$(C)=A$(C)
1040        NEXT C
1050        PRINT
1060        PRINT LIN(2);Bcount
1070        FOR C=1 TO Nv
1080           IF A$(C)<>"+" THEN 1140
1090           GOTO 1170
1100           X(Nobs,C)=VAL(B$(C))
1110           A$(C)=B$(C)
1120           PRINT USING 810;X(Nobs,C)
1130           GOTO 1170
1140           IF A$(C)="+" THEN 1170
1150           X(Nobs,C)=VAL(A$(C))
1160           PRINT USING 810;X(Nobs,C)
1170           B$(C)=A$(C)
1180        NEXT C
1190     GOTO 970
1200 Subfi:INPUT "ANOTHER SUBFILE?",S$
1210     GOSUB Check
1220     IF Check=1 THEN Subfi
1230     Nobs=Nobs-1
1240     Subf(Nsub+1)=Nobs
1250     PRINT CHR$(27)&"m",PAGE
1260     IF UPC$(S$[1,1])="Y" THEN 840
1270  INPUT "STORE DATA?",S$
1280  IF UPC$(S$[1,1])="N" THEN 1360
1290  Phyrec=(Nobs*20*12/256+Nsub*32/256+2)*3/2
1300  REDIM A$(Nv),B$(Nv),Nsub$(Nsub),Subf(Nsub+1),X(Nobs,Nv)
1310  INPUT "FILE NAME?",Filnm$
1320  CREATE Filnm$,Phyrec
1330  ASSIGN #1 TO Filnm$
1340  PRINT #1;Nv,Nobs,Nsub
1350  PRINT #1;Nsub$(*),Subf(*),X(*)
1360 Restart:  Flaggg=0
1361  ON KEY #1 GOTO 1450
1370  ON KEY #2 GOTO 1470
1380  ON KEY #3 GOTO 1490
1390  ON KEY #4 GOTO 1290
1400  ON KEY #5 GOTO 1510
1410  ON KEY #6 GOTO 1440
1420  DISP "SELECT K1-K6"
1421  Flaggg=Flaggg+1
1430  GOTO 1430
1440  END
1450  DISP "GOING TO REGRESSION PROGRAM."
1460  LOAD "REGRES:T15"
1470  IF Flaggg>1 THEN GOTO 1479
1471  DISP "PLOTS CAN ONLY BE ACCESSED AFTER REGRESSION HAS BEEN RUN"
1472  WAIT 6000
1473  DISP "SELECT k1-K6"
1474  GOTO 1430
1479  DISP "GOING TO PLOTTING PROGRAM."
1480  LOAD "PLOTS:T15"
1490  DISP "GOING TO EDIT PROGRAM."
1500  LOAD "EDIT:T15"
1510  DISP "GOING TO LISTING PROGRAM."
1520  LOAD "LIST:T15"
1530 Check: Check=0
1540  PRINT PAGE
1550  IF (UPC$(S$[1,1])="Y") OR (UPC$(S$[1,1])="N") THEN 1590
1560  Check=1
1570  BEEP
1580  PRINT PAGE;"*** IMPROPER RESPONSE - TRY AGAIN ***"
1590  RETURN
1600  ! CHECKS FILENAME
1610 Badfile: BEEP
```

```
1620  IF ERRN<>56 THEN 1650
1630  PRINT PAGE;"*** FILE NAME DOES NOT EXIST ***"
1640  GOTO 690
1650  DISP ERRM$
1660  GOTO 1440
```

```
10    REM --> PROGRAM REGRES
20    OPTION BASE 1
30    COM X(500,20),V$(20)[160],Nsub$(20)[10],Ab(20,21),INTEGER Dep,Subf(20),Nsu
b,Nobs,Nv,Nt,M$(20)[150],Term
32    REDIM X(Nobs,Nv)
40    DIM Dp$[160],St$[160],St1$[160],Sl(20),Keep(20),N$(20)[150],Test$[150],Pr$
[150]
50    INTEGER User(20)
60    MAT Ab=ZER
70    Slct=1
80    Kterm=0
90    DISP
100   Last=N=Nt
110   IF V$(1)="" THEN 370
120   Last=N=Nt
140   INPUT "USE SAME EQUATION?",S$
150   GOSUB Check
160   IF Check=1 THEN 140
170   IF UPC$(S$[1,1])="Y" THEN 2340
180   INPUT "DO YOU WANT TO KEEP ANY OF THE TERMS OF THE EQUATION?",S$
190   GOSUB Check
200   IF Check=1 THEN 180
210   IF UPC$(S$[1,1])="N" THEN 670
220   INPUT "HOW MANY TERMS DO YOU WANT TO KEEP?",Knt
230   IF (Knt>0) AND (Knt<=20) AND (FRACT(Knt)=0) THEN 270
240   BEEP
250   PRINT PAGE;"*** IMPROPER NUMBER OF TERMS ***"
260   GOTO 220
270   PRINT PAGE
280   FOR A=1 TO Knt
290      INPUT "TERM #?",Keep(A)
300      IF (Keep(A)>0) AND (Keep(A)<=20) AND (FRACT(Keep(A))=0) THEN 340
310      BEEP
320      PRINT PAGE;"*** IMPROPER TERM NUMBER ***"
330      GOTO 290
340      PRINT PAGE
350   NEXT A
360   GOTO 670
370   PRINT PAGE,TAB(25);CHR$(129);"NOTES ON REGRESSION PROGRAM";CHR$(128)
380   PRINT LIN(2),TAB(5);"The regression program takes an equation defined by t
he user such as:"
390   PRINT LIN(1);"           Y=a*X^3+b*X*ln(Z)+c"
400   PRINT LIN(1);"and solves for a,b,and c based on the best fit to the curren
t data."
410   PRINT LIN(1),TAB(5);"The relational expressions are the input terms of the
 regression equation."
420   PRINT "Once you have input the regression equation, you can perform the "
430   PRINT "regression on any combination of subfiles to find the relation in o
ne "
440   PRINT "particular subfile.  Later, even after a plot has been performed, y
ou can "
450   PRINT "repeat the same regression without the need of retyping the regress
ion equation"
```

```
460   PRINT "or the user defined terms.  So you can perform the regression on on
ly a few"
470   PRINT "subfiles the first time, and later perform the regression on the wh
ole data set."
480   PRINT "Also, you can pick and choose which regression terms and which user
 defined "
490   PRINT "terms you wish to keep, so remember the order in which you input th
e terms to "
500   PRINT "capitalize on this feature."
510   DISP "PRESS 'CONT' TO CONTINUE."
520   PAUSE
530   PRINT PAGE,TAB(20);CHR$(129);"HOW TO INPUT USER DEFINED TERMS";CHR$(128)
540   PRINT LIN(2),TAB(5);"User defined terms are relational expressions that
complement the "
550   PRINT "regression equation by defining constants that can be used in the e
quation."
560   PRINT "These terms are independent equations that must have proper syntax,
 and any"
570   PRINT "variable names can be used except A,B, and any variable in common.
User defined "
580   PRINT "terms are placed, in order, before the regression equation, and are
 evaluated"
590   PRINT "for every point in the data set.  It is important that the syntax o
f any user"
600   PRINT "input terms is correct.  If, for any reason, the program gets lost"
610   PRINT "(i.e. the screen is blank, and the tape drive is not working) for a
n extended"
620   PRINT "length of time, stop the program, and start over.  The problem was
due to "
630   PRINT "improper syntax of the input equations."
640   PRINT LIN(1),TAB(5);"User defined terms can also be used with any programa
ble function"
650   PRINT "such as PRINT or RAD to display any intermediate results or tempora
rily set the"
660   PRINT "computer into a desired computational mode."
670   IF M$(1)="" THEN 880
680   INPUT "DO YOU WANT TO KEEP ANY OF THE USER DEFINED TERMS?",S$
690   GOSUB Check
700   IF Check=1 THEN 680
710   IF UPC$(S$[1,1])="N" THEN 880
720   INPUT "HOW MANY USER DEFINED TERMS DO YOU WANT TO KEEP?",Kterm
730   IF (Kterm>0) AND (Kterm<20) AND (FRACT(Kterm)=0) THEN 770
740   BEEP
750   PRINT PAGE;"*** IMPROPER NUMBER OF USER DEFINED TERMS ***"
760   GOTO 720
770   PRINT PAGE
780   FOR A=1 TO Kterm
790      INPUT "KEEP USER DEFINED #",User(A)
800      IF (User(A)>0) AND (User(A)<=20) AND (FRACT(User(A))=0) THEN 840
810      BEEP
820      PRINT PAGE;"*** IMPROPER USER DEFINED TERM NUMBER ***"
830      GOTO 790
840      PRINT PAGE
850      N$(User(A))=M$(User(A))
860   NEXT A
870   GOTO 930
880   INPUT "DO YOU WANT ANY USER DEFINED TERMS?",S$
890   GOSUB Check
900   IF Check=1 THEN 880
910   Term=0
920   IF UPC$(S$[1,1])="N" THEN 1140
930   INPUT "HOW MANY USER DEFINED TERMS?",Term
940   IF (Term>0) AND (Term<=20) AND (FRACT(Term)=0) THEN 980
950   PRINT PAGE;"*** IMPROPER NUMBER OF USER DEFINED TERMS ***"
960   BEEP
970   GOTO 930
980   PRINT PAGE
990   FOR A=1 TO Term
```

```
1000      Flag=0
1010      FOR B=1 TO Kterm
1020         IF User(B)=A THEN Flag=1
1030      NEXT B
1040      IF Flag<>1 THEN 1070
1050      N$(A)=M$(A)
1060      GOTO 1090
1070      DISP "INPUT USER DEFINED TERM #";A;"  (IN QUOTES;150 character maximum)
";
1080      INPUT "",N$(A)
1090      Test$=N$(A)
1100      GOSUB Parenth
1110      IF Ptest=1 THEN 1070
1120      M$(A)=N$(A)
1130   NEXT A
1140   PRINT PAGE,TAB(13);CHR$(129);"PROCEDURE FOR THE INPUT OF THE REGRESSION EQ
UATION";CHR$(128)
1150   PRINT LIN(4);CHR$(132);"FOR THE EQUATION  'X(A,3)=I*X(A,2)^2 + J*X(A,1) +
K'";CHR$(128)
1160   PRINT LIN(1);"   X(A,3) = THE DEPENDENT VARIABLE. IN THIS CASE THE THIRD V
ARIABLE."
1170   PRINT LIN(1);"   I*X(A,2)^2 = THE FIRST TERM OF THE EQUATION FOR WHICH  I
 IS TO BE SOLVED."
1180   PRINT "      (Input 'X(A,2)^2' as the first term within double quotes.)"
1190   PRINT LIN(1);"   J*X(A,1) = THE SECOND TERM FOR WHICH J IS TO BE SOLVED."
1200   PRINT LIN(1);"   K = THE THIRD TERM WHICH IS A CONSTANT WITHOUT A VARIABLE
."
1210   PRINT "      (Input '1' to represent this term.)"
1220   PRINT LIN(1);"Regression terms are a maximum of 70 characters in length."
1230 Start:INPUT "HOW MANY TERMS ARE THERE IN THE REGRESSION?",Nt
1240   IF (Nt>0) AND (Nt<=20) AND (FRACT(Nt)=0) THEN 1280
1250   BEEP
1260   PRINT PAGE;"*** IMPROPER NUMBER OF TERMS ***"
1270   GOTO Start
1280   PRINT PAGE
1290   REDIM Ab(Nt,Nt+1)
1300   PRINT
1310   Last=N=Nt
1330   INPUT "INPUT THE INDEX OF THE DEPENDENT VARIABLE. (e.g. 1 the for 1st vari
ble)",Dep
1340   IF (Dep>0) AND (Dep<=Nv) AND (FRACT(Dep)=0) THEN 1380
1350   BEEP
1360   PRINT PAGE;"*** IMPROPER VARIABLE NUMBER ***"
1370   GOTO 1330
1380   PRINT PAGE
1390   V$(Nt+1)="X(A,"&VAL$(Dep)&")"
1400   FOR A=1 TO Nt
1410      Flag=0
1420      FOR B=1 TO Knt
1430         IF A=Keep(B) THEN Flag=1
1440      NEXT B
1450      IF Flag=1 THEN 1590
1460      DISP "INPUT TERM #";A;" (e.g.  X(A,1) within quotes)";
1470      INPUT "",V$(A)
1480      Test$=V$(A)
1490      GOSUB Parenth
1500      IF Ptest=1 THEN 1460
1510      St=POS(V$(A),V$(Nt+1))
1520      IF St=0 THEN 1590
1530      PRINT USING "+,K";V$(A);" IS NOT VALID SINCE THE DEPENDENT VARIABLE IS
";V$(Nt+1)
1540      BEEP
1550      DISP "PRESS CONT TO CONTINUE"
1560      PAUSE
1570      PRINT USING "+,K";"
                             "
1580      GOTO 1460
1590   NEXT A
```

```
1600  PRINT
1610  ON ERROR GOTO 1630
1620  GOTO 1680
1630  IF ERRN<>54 THEN 1660
1640  PURGE "BUFFER:T15"
1650  GOTO 1680
1660  PRINT ERRM$
1670  PAUSE
1680  CREATE "BUFFER:T15",(Nt*(Nt+Nv+Term+3)+5)*80/256+1
1690  ASSIGN "BUFFER:T15" TO #1
1700  PRINT #1;"50  Builder:!"
1710  PRINT #1;"51  FOR B=1 TO Fx"
1720  Ln=50
1730  PRINT #1;VAL$(51+Ln)&"  FOR A=Subf(S1(B))+1 TO Subf(S1(B)+1)"
1740  Ln=Ln+1
1750  FOR A=1 TO Term
1760     PRINT #1;VAL$(51+Ln)&"  "&N$(A)
1770     Ln=Ln+1
1780     Po1=POS(N$(A),"X(A,")
1790     IF Po1=0 THEN Poend
1800     Po2=Po1+POS(N$(A)[Po1],")")
1810     N$(A)[Po1,Po2-1]="S1("&N$(A)[Po1+4,Po2-2]&")"
1820     GOTO 1780
1830 Poend:    NEXT A
1840  FOR A=1 TO Nt
1850     FOR B=1 TO Nt+1
1860        St$=VAL$(51+Ln)&"  Ab("&VAL$(A)&","&VAL$(B)&")="&"Ab("&VAL$(A)&","&V
AL$(B)&")+"&V$(A)&"*"&V$(B)
1870        Ln=Ln+1
1880        PRINT #1;St$
1890     NEXT B
1900  NEXT A
1910  PRINT #1;VAL$(51+Ln)&"  NEXT A"
1920  PRINT #1;VAL$(52+Ln)&"  NEXT B"
1930  PRINT #1;VAL$(53+Ln)&"  GOTO 5500 "
1940  Ln=Ln+3
1950  FOR D=1 TO Nv
1960     FOR B=1 TO Nt
1970        St$="  Fx=Fx+"&V$(B)&"+Ab("&VAL$(B)&","&VAL$(Nt+1)&")"
1980        St1$=St$
1990        Strt=POS(St1$,"X(A,"&VAL$(D))
2000        IF Strt=0 THEN 2040
2010        Nd=POS(St1$[Strt],")")
2020        St1$[Strt,Strt+Nd-1]="Var"
2030        GOTO 1990
2040        FOR C=1 TO Nv
2050           Str=POS(St1$,"X(A,"&VAL$(C))
2060           IF Str=0 THEN 2100
2070           En=POS(St1$[Str],")")
2080           St1$[Str,Str+En-1]="S1("&VAL$(C)&")"
2090           GOTO 2050
2100        NEXT C
2110        IF B<>1 THEN 2240
2120        PRINT #1;VAL$(51+Ln)&" A"&VAL$(D)&":  Fx=0"
2130        Ln=Ln+1
2140        FOR Ed=1 TO Term
2150           Pr$=N$(Ed)
2160           Po1=POS(Pr$,"S1("&VAL$(D))
2170           IF Po1=0 THEN 2210
2180           Po2=Po1+POS(Pr$[Po1],")")
2190           Pr$[Po1,Po2]="Var"
2200           GOTO 2160
2210           PRINT #1;VAL$(51+Ln)&"  "&Pr$
2220           Ln=Ln+1
2230        NEXT Ed
2240        PRINT #1;VAL$(51+Ln)&"  "&St1$
2250        Ln=Ln+1
2260     NEXT B
2270     PRINT #1;VAL$(51+Ln)&"  GOTO 5500"
```

```
2280     Ln=Ln+1
2290     St1$=St$
2300  NEXT D
2310  PRINT #1;"5500  SUBEND"
2320  ASSIGN #1 TO *
2330  OFF ERROR
2340  LINK "BUFFER:T15",3450,2350
2350  PRINT PAGE
2360  INPUT "DO YOU WANT TO DO THE REGRESSION ON ALL SUBFILES?",S$
2370  GOSUB Check
2380  IF Check=1 THEN 2360
2390  IF UPC$(S$[1,1])="Y" THEN 2560
2400  INPUT "HOW MANY SUBFILES DO YOU WANT INCLUDED IN THE REGRESSION?",Fx
2410  IF (Fx>0) AND (Fx<=Nsub) AND (FRACT(Fx)=0) THEN 2460
2420  DISP "*** WARNING - NUMBER OF SUBFILES EXCEEDED ***"
2430  BEEP
2440  WAIT 2000
2450  GOTO 2400
2460  FOR A=1 TO Fx
2470     DISP "INPUT SUBFILE #";A;" FOR THE REGRESSION.";
2480     INPUT "",S1(A)
2490     IF (S1(A)>0) AND (S1(A)<=Nsub) AND (FRACT(S1(A))=0) THEN 2530
2500     BEEP
2510     PRINT PAGE;"*** IMPROPER SUBFILE NUMBER ***"
2520     GOTO 2470
2530     PRINT PAGE
2540  NEXT A
2541  PRINT PAGE,LIN(8),TAB(25);CHR$(129);"REGRESSION IN PROGRESS";CHR$(128)
2550  GOTO 2610
2560  FOR A=1 TO Nsub
2570     S1(A)=A
2580  NEXT A
2590  PRINT PAGE,LIN(8),TAB(25);CHR$(129);"REGRESSION IN PROGRESS";CHR$(128)
2600  Fx=Nsub
2610  CALL Build(Slct,Var,Fx,S1(*),N$(*))
2620           ! SIMULTANEOUS EQUATION PROGRAM OF Nth ORDER.
2630  FOR Mat=1 TO N-1
2640     First=Mat
2650     Large=First
2660     FOR Test=First+1 TO Last
2670        IF Ab(Test,First)>Ab(Large,First) THEN Large=Test
2680     NEXT Test
2690     IF Large=First THEN GOTO 2750
2700     FOR Switch=First TO Last+1
2710        Save=Ab(First,Switch)
2720        Ab(First,Switch)=Ab(Large,Switch)
2730        Ab(Large,Switch)=Save
2740     NEXT Switch
2750     FOR Diag=First+1 TO Last
2760        Ratio=Ab(Diag,First)/Ab(First,First)
2770        FOR Zero=First TO Last+1
2780           Ab(Diag,Zero)=Ab(Diag,Zero)-Ratio*Ab(First,Zero)
2790        NEXT Zero
2800     NEXT Diag
2810  NEXT Mat
2820  FOR Solve=Last TO 1 STEP -1
2830     Ab(Solve,Last+1)=Ab(Solve,Last+1)/Ab(Solve,Solve)
2840     IF Solve=1 THEN GOTO 2890
2850     FOR Solv=Solve-1 TO 1 STEP -1
2860        Ab(Solv,Solve)=Ab(Solve,Last+1)*Ab(Solv,Solve)
2870        Ab(Solv,Last+1)=Ab(Solv,Last+1)-Ab(Solv,Solve)
2880     NEXT Solv
2890  NEXT Solve
2900  PRINT PAGE
2910  PRINT LIN(3)
2920  PRINT USING "#,K";"Fx=a*";V$(1)
2930  FOR A=2 TO Nt
2940     PRINT USING "#,K";"+";CHR$(96+A);"*";V$(A)
```

```
2950  NEXT A
2960  PRINT
2970  FOR A=1 TO Nt
2980     PRINT TAB(3);CHR$(96+A);"=";Ab(A,Nt+1)
2990  NEXT A
3000  FOR A=1 TO Term
3010     PRINT LIN(1);M$(A)
3020  NEXT A
3021  BEEP
3022  WAIT 1000
3023  BEEP
3030  INPUT "DO YOU WANT A PLOT?",S$
3040  GOSUB Check
3050  IF Check=1 THEN 3030
3060  IF UPC$(S$[1,1])="Y" THEN LOAD "PLOTS:T15"
3070  PRINT PAGE
3080  DISP "RETURNING TO MAIN PROGRAM"
3090  LOAD "AUTOST:T15",Restart
3100  !
3110  ! PARENTHESIS CHECKING ROUTINE
3120 Parenth: Ptest=0
3130  Left_count=Right_count=Poss=Po=0
3140  Po=POS(Test$,"(")
3150  IF Po=0 THEN Leftpas
3160  Test$[Po,Po]="*"
3170  Left_count=Left_count+1
3180  GOTO 3140
3190 Leftpas: Poss=POS(Test$,")")
3200  IF Poss=0 THEN Rightpas
3210  Test$[Poss,Poss]="*"
3220  Right_count=Right_count+1
3230  GOTO Leftpas
3240 Rightpas: IF Left_count=Right_count THEN RETURN
3250  Ptest=1
3260  IF Left_count>Right_count THEN DISP Left_count-Right_count;"EXCESS LEFT PA
RENTHESIS"
3270  IF Left_count<Right_count THEN DISP Right_count-Left_count;"EXCESS RIGHT P
ARENTHESIS"
3280  BEEP
3290  WAIT 3000
3300  RETURN
3310  !
3320  ! CHECKS YES AND NO RESPONSES
3330 Check: Check=0
3340  PRINT PAGE
3350  IF (UPC$(S$[1,1])="Y") OR (UPC$(S$[1,1])="N") THEN RETURN
3360  Check=1
3370  BEEP
3380  PRINT PAGE;"*** IMPPOPER RESPONSE - TRY AGAIN."
3390  RETURN
3400  !
3410  SUB Build(Slct,Var,Fx,Sl(*),N$(*))
3420  OPTION BASE 1
3430  COM X(*),V$(*),Nsub$(*),Ab(*),INTEGER Dep,Subf(*),Nsub,Nobs,Nv,Nt
3440  ON Slct GOTO Builder,A1,A2,A3,A4,A5,A6,A7,A8,A9,A10,A11,A12,A13,A14,A15,A1
6,A17,A18,A19,A20
3450 Builder:!
```

```
10    REM --> PROGRAM LIST
20    OPTION BASE 1
30    COM X(500,20),V$(20)[160],Nsub$(20)[10],Ab(20,21),INTEGER Dep,Subf(20),Nsu
b,Nobs,Nv,Nt,M$(20)[150],Term
```

```
32    REDIM X(Nobs,Nv)
40    DIM Sub(20)
50    INPUT "DO YOU WANT ALL SUBFILES LISTED?",S$
60    IF (UPC$(S$[1,1])="Y") OR (UPC$(S$[1,1])="N") THEN 91
70    BEEP
80    PRINT PAGE;"*** IMPROPER RESPONSE - TRY AGAIN ***"
90    GOTO 50
91    INPUT "DO YOU WANT A HARD COPY (H) OR A CRT DISPLAY (C)?",T$
92    IF (UPC$(T$[1,1])="H") OR (UPC$(T$[1,1])="C") THEN 96
93    BEEP
94    PRINT PAGE;"  *** IMPROPER RESPONCE, PLEASE TRY AGAIN ***"
95    GOTO 91
96    IF UPC$(T$[1,1])="C" THEN 101
97    PRINTER IS 0
101   PRINT PAGE
110   IF UPC$(S$[1,1])="Y" THEN 280
120   INPUT "HOW MANY SUBFILES DO YOU WANT LISTED?",Nss
130   IF (Nss>0) AND (Nss<=Nsub) AND (FRACT(Nss)=0) THEN 170
140   BEEP
150   PRINT PAGE;"*** IMPROPER NUMBER OF SUBFILES ***"
160   GOTO 120
170   PRINT PAGE
180   FOR A=1 TO Nss
190      DISP "INPUT SUBFILE #";A;
200      INPUT "",Sub(A)
210      IF (Sub(A)>0) AND (Sub(A)<=Nsub) AND (FRACT(Sub(A))=0) THEN 250
220      BEEP
230      PRINT PAGE;"*** IMPROPER SUBFILE NUMBER ***"
240      GOTO 190
250      PRINT PAGE
260   NEXT A
270   GOTO 320
280   FOR A=1 TO Nsub
290      Sub(A)=A
300   NEXT A
310   Nss=Nsub
340   FOR A=1 TO Nss
341      PRINT LIN(3)
350      PRINT LIN(1);"SUBFILE: ",Nsub$(A)
360      PRINT LIN(1)
370      Jj=0
380      FOR Ii=1 TO Nv
390      Jj=Jj+1
400      IF Jj<=5 THEN GOTO 430
410      Jj=1
420      PRINT USING 460
430      PRINT USING 450;"VAR#"&VAL$(Ii)
440      NEXT Ii
450      IMAGE #,7X,6A
460      IMAGE /
470      PRINT USING 460
480      PRINT " OBS #"
490      Obs=0
500      FOR B=Subf(Sub(A))+1 TO Subf(Sub(A)+1)
510         Obs=Obs+1
520         PRINT LIN(1)
530         PRINT Obs
540           Jj=0
550           FOR Ii=1 TO Nv
560             Jj=Jj+1
570             IF Jj<=5 THEN GOTO 600
580             Jj=1
590             PRINT USING 460
600             PRINT USING 680;X(B,Ii)
610           NEXT Ii
620      NEXT B
630   NEXT A
631   PRINT LIN(2)
```

```
640   PRINTER IS 16
650   PAUSE
660   DISP "RETURNING TO MAIN PROGRAM"
670   LOAD "AUTOST:T15",Restart
680   IMAGE #,5X,5D.4D
690   IMAGE 10X
```

```
10    REM --> PROGRAM PLOTS
20    OPTION BASE 1
30    COM X(500,20),V$(20)[160],Nsub$(20)[10],Ab(20,21),INTEGER Dep,Subf(20),Nsu
b,Nobs,Nv,Nt,M$(20)[150],Term
32    REDIM X(Nobs,Nv)
40    DIM Gxmxx(20),Gxmnx(20),Gymxx(20),Gymnx(20),Sl(20),Eq$[500],Xla$[160],Yla$
[160]
60    INTEGER Nfile(20)
70    PRINT PAGE,TAB(25);CHR$(129);"NOTES ON PLOTTING:";CHR$(128)
80    PRINT LIN(2),TAB(5);"This program will plot the data and the regression fu
nction using any"
90    PRINT "variable as the independent variable except the dependent variable.
"
100   PRINT LIN(1),TAB(5);"The function can be plotted using a line and setting
all variables"
110   PRINT "to a constant except the independent and dependent variables.  Mult
iple lines"
120   PRINT "can be drawn in this manner with the option of changing any of the
constants."
130   PRINT "Another option is to plot corresponding data points that inserts al
l variables"
140   PRINT "from each data point into the equation."
150   PRINT LIN(1),TAB(5);"Any combination of subfiles can be plotted."
160   LINK "BUFFER:T15",2340,170
170   GCLEAR
180   EXIT GRAPHICS
190   DEG
200   INPUT "INDEPENDENT VARIABLE?",Ind
210   IF Ind<>Dep THEN 240
220   PRINT PAGE,LIN(7);"THAT IS THE DEPENDENT VARIABLE."
221   GOTO 200
230   GOTO 200
240   PRINT PAGE
250   INPUT "PLOT THE COMPLETE DATA SET?",S$
260   GOSUB Check
270   IF Check=1 THEN 250
280   IF UPC$(S$[1,1])="N" THEN 380
290   Sbsts=Nsub
300   MAT SEARCH X(*,Ind),MIN;Xmn
310   MAT SEARCH X(*,Ind),MAX;Xmx
320   MAT SEARCH X(*,Dep),MIN;Ymn
330   MAT SEARCH X(*,Dep),MAX;Ymx
340   FOR A=1 TO Nsub
350      Nfile(A)=A
360   NEXT A
370   GOTO 650
380   INPUT "HOW MANY SUB FILES?",Sbsts
390   IF (Sbsts>0) AND (Sbsts<=Nsub) AND (FRACT(Sbsts)=0) THEN 430
400   PRINT PAGE;"*** IMPROPER NUMBER OF SUBFILES ***"
410   BEEP
420   GOTO 380
430   PRINT PAGE
440   REDIM Gxmnx(Sbsts),Gxmxx(Sbsts),Gymnx(Sbsts),Gymxx(Sbsts)
450   FOR A=1 TO Sbsts
```

```
460      INPUT "NUMBER OF SUBFILE?",Nfile(A)
470      IF (Nfile(A)>0) AND (Nfile(A)<=Nsub) AND (FRACT(Nfile(A))=0) THEN 510
480      BEEP
490      PRINT PAGE;"*** IMPROPER SUBFILE NUMBER ***"
500      GOTO 460
510      PRINT PAGE
520      Gxmnx(A)=Gxmxx(A)=X(Subf(Nfile(A))+1,Ind)
530      Gymnx(A)=Gymxx(A)=X(Subf(Nfile(A))+1,Dep)
540      FOR B=Subf(Nfile(A))+1 TO Subf(Nfile(A)+1)
550         IF X(B,Ind)>Gxmxx(A) THEN Gxmxx(A)=X(B,Ind)
560         IF X(B,Ind)<Gxmnx(A) THEN Gxmnx(A)=X(B,Ind)
570         IF X(B,Dep)>Gymxx(A) THEN Gymxx(A)=X(B,Dep)
580         IF X(B,Dep)<Gymnx(A) THEN Gymnx(A)=X(B,Dep)
590      NEXT B
600   NEXT A
610   MAT SEARCH Gxmxx,MAX;Xmx
620   MAT SEARCH Gxmnx,MIN;Xmn
630   MAT SEARCH Gymxx,MAX;Ymx
640   MAT SEARCH Gymnx,MIN;Ymn
650   PRINT PAGE;"X-MIN = ";DROUND(Xmn,3);"        XMAX = ";DROUND(Xmx,3)
660   PRINT LIN(2);"Y-MIN = ";DROUND(Ymn,3);"        YMAX = ";DROUND(Ymx,3)
670   INPUT "INPUT GRAPH LIMITS FOR X-MIN AND X-MAX.",Gxmn,Gxmx
680   INPUT "INPUT GRAPH LIMITS FOR Y-MIN AND Y-MAX.",Gymn,Gymx
690   PRINT PAGE
700   LIMIT 0,184.47,0,149.8
710   LOCATE 15,120,15,95
720   CSIZE 3.38
730   SCALE Gxmn,Gxmx,Gymn,Gymx
740   FRAME
750   DISP "HANG ON A MINUTE"
760   Xtic=(Gxmx-Gxmn)/10
770   Ytic=(Gymx-Gymn)/10
780   AXES Xtic,Ytic,Gxmn,Gymn
790   LORG 8
800   LDIR 90
810   FOR A=Gxmn TO Gxmx STEP Xtic
820      MOVE A,Gymn
830      LABEL DROUND(A,3)
840   NEXT A
850   INPUT "X-AXIS LABEL",Xla$
860   DISP "HANG ON A MINUTE"
870   LDIR 0
880   FOR A=Gymn TO Gymx STEP Ytic
890      MOVE Gxmn,A
900      LABEL DROUND(A,3)
910   NEXT A
920   LORG 6
930   MOVE (Gxmx+Gxmn)/2,Gymn-(Gymx-Gymn)/8
940   LABEL Xla$
950   INPUT "Y-AXIS LABEL",Yla$
960   MOVE Gxmn-(Gxmx-Gxmn)/8.1,(Gymx+Gymn)/2
970   LORG 4
980   LDIR 90
990   LABEL Yla$
1000  LORG 5
1010  INPUT "DO YOU WANT AN EQUATION LINE(1) OR CORRESPONDING DATA POINTS(0)?",S
1020  IF (S=0) OR (S=1) THEN 1060
1030  BEEP
1040  PRINT PAGE;"*** IMPROPER RESPONSE - TRY AGAIN ***"
1050  GOTO 1010
1060  PRINT PAGE
1070  Slct=Ind+1
1080  IF S=0 THEN 1430
1090  Pass=0
1100  IF Pass<>0 THEN 1120
1110  GOTO 1370
1120  INPUT "DO YOU WANT THE SAME PARAMETERS?",S$
1130  GOSUB Check
1140  IF Check=1 THEN 1120
```

```
1150  IF UPC$(S$[1,1])="Y" THEN 1430
1160  INPUT "HOW MANY PARAMETERS DO YOU WANT TO CHANGE?",S
1170  IF (S>0) AND (S<=Nv) AND (FRACT(S)=0) THEN 1210
1180  PRINT PAGE;"*** IMPROPER NUMBER OF PARAMETERS ***"
1190  BEEP
1200  GOTO 1160
1210  PRINT PAGE
1220  FOR A=1 TO S
1230     INPUT "WHAT VARIABLE NUMBER?",Vnn
1240     IF (Vnn>0) AND (Vnn<=Nv) AND (FRACT(Vnn)=0) THEN 1280
1250     BEEP
1260     PRINT PAGE;"*** IMPROPER VARIABLE NUMBER ***"
1270     GOTO 1230
1280     PRINT PAGE
1290     IF (Vnn<>Dep) AND (Vnn<>Ind) THEN 1320
1300     PRINT PAGE,LIN(7);"THAT IS EITHER THE INDEPENDENT OR THE DEPENDENT VARI
ABLE."
1310     GOTO 1230
1320     PRINT PAGE
1330     DISP "WHAT VALUE DO YOU WANT FOR VARIABLE #";Vnn;
1340     INPUT "",S1(Vnn)
1350  NEXT A
1360  GOTO 1420
1370  FOR A=1 TO Nv
1380     IF (A=Ind) OR (A=Dep) THEN 1410
1390     DISP "WHAT VALUE DO YOU WANT FOR VARIABLE #";A;
1400     INPUT "",S1(A)
1410  NEXT A
1420  Pass=1
1430  GRAPHICS
1440  Variance=Nbcount=0
1450  FOR A=1 TO Sbsts
1460     FOR B=Subf(Nfile(A))+1 TO Subf(Nfile(A)+1)
1470        Nbcount=Nbcount+1
1480        MOVE X(B,Ind),X(B,Dep)
1490        LABEL "*"
1500        Var=X(B,Ind)
1510        FOR C=1 TO Nv
1520          IF (C=Dep) OR (C=Ind) THEN 1540
1530          S1(C)=X(B,C)
1540        NEXT C
1550        CALL Build(Slct,Var,Fx,S1(*))
1560        Variance=Variance+(Fx-X(B,Dep))^2
1570        IF S<>0 THEN 1600
1580        MOVE Var,Fx
1590        LABEL "O"
1600     NEXT B
1610  NEXT A
1620  IF S<>0 THEN 1710
1630  LORG 2
1640  MOVE Gxmn+Xtic/3,Gymx-Ytic/3
1650  CSIZE 2.7
1660  LDIR 0
1670  LABEL "O = CALCULATED"
1680  MOVE Gxmn+Xtic/3,Gymx-3/4*Ytic
1690  LABEL "* = DATA"
1700  GOTO 1840
1710  FOR A=Gxmn TO Gxmx STEP (Gxmx-Gxmn)/50
1720     Var=A
1730     CALL Build(Slct,Var,Fx,S1(*))
1740     IF A=Gxmn THEN MOVE Var,Fx
1750     PLOT Var,Fx
1760  NEXT A
1770  PAUSE
1780  EXIT GRAPHICS
1790  INPUT "ANOTHER LINE?",S$
1800  GOSUB Check
1810  IF Check=1 THEN 1790
1820  IF UPC$(S$[1,1])="Y" THEN 1120
```

```
1830  GOTO 1850
1840  PAUSE
1850  INPUT "DO YOU WANT A HARD COPY?",S$
1860  GOSUB Check
1870  IF Check=1 THEN 1850
1880  IF UPC$(S$[1,1])="N" THEN 2050
1890  PRINTER IS 0
1900  PRINT PAGE
1910  DUMP GRAPHICS
1920  Eq$="Fx=a*"&V$(1)
1930  FOR A=2 TO Nt
1940     Eq$=Eq$&"+"&CHR$(96+A)&"*"&V$(A)
1950  NEXT A
1960  PRINT Eq$
1970  FOR A=1 TO Nt
1980     PRINT CHR$(A+96)&"="&VAL$(Ab(A,Nt+1))
1990  NEXT A
2000  FOR A=1 TO Term
2010     PRINT LIN(1);M$(A)
2020  NEXT A
2030  PRINT LIN(1);"Goodness-of fit is  ";PROUND(Variance/Nbcount,-3)
2040  PRINTER IS 16
2050  INPUT "ANOTHER PLOT?",S$
2060  GOSUB Check
2070  IF Check=1 THEN 2050
2080  IF UPC$(S$[1,1])="Y" THEN 170
2090  PRINT PAGE
2100  GCLEAR
2110  EXIT GRAPHICS
2120  INPUT "ANOTHER REGRESSION?",S$
2130  GOSUB Check
2140  IF Check=1 THEN 2120
2150  IF UPC$(S$[1,1])="Y" THEN DISP "GOING TO REGRESSION PROGRAM."
2160  IF UPC$(S$[1,1])="Y" THEN LOAD "REGRES:T15"
2170  DISP "RETURNING TO MAIN PROGRAM"
2180  LOAD "AUTOST:T15",Restart
2190  !
2200  ! CHECKS YES AND NO RESPONSES
2210 Check: Check=0
2220  PRINT PAGE
2230  IF (UPC$(S$[1,1])="Y") OR (UPC$(S$[1,1])="N") THEN 2270
2240  BEEP
2250  Check=1
2260  PRINT PAGE;"*** IMPROPER RESPONSE - TRY AGAIN ***"
2270  RETURN
2280  !
2290  !
2300  SUB Build(Slct,Var,Fx,Sl(*))
2310  OPTION BASE 1
2320  COM X(*),V$(*),Nsub$(*),Ab(*),INTEGER Dep,Subf(*),Nsub,Nobs,Nv,Nt,M$(*),Te
rm
2330  ON Slct GOTO Builder,A1,A2,A3,A4,A5,A6,A7,A8,A9,A10,A11,A12,A13,A14,A15,A1
6,A17,A18,A19,A20
2340 Builder:!
```

```
10     REM --> PROGRAM EDIT
20     OPTION BASE 1
30     COM X(500,20),V$(20)[160],Nsub$(20)[10],Ab(20,21),INTEGER Dep,Subf(20),Nsu
b,Nobs,Nv,Nt,M$(20)[150],Term
31     REDIM X(500,Nv)
40 Menu:    PRINT PAGE,LIN(3);"SELECT AN EDITOR FUNCTION:"
50     PRINT LIN(1);"   1) CHANGE AN OBSERVATION."
60     PRINT LIN(1);"   2) CHANGE A SET OF OBSERVATIONS."
```

```
70    PRINT LIN(1);"    3) ADD A SUBFILE."
80    PRINT LIN(1);"    4) ADD AN OBSERVATION."
90    PRINT LIN(1);"    5) DELETE AN OBSERVATION."
100   PRINT LIN(1);"    6) MERGE A DATA FILE."
110   PRINT LIN(1);"    7) QUIT THE EDITOR."
120   INPUT "",Ef
130   IF (Ef>0) AND (Ef<8) AND (FRACT(Ef)=0) THEN 180
140   BEEP
150   PRINT PAGE;"*** IMPROPER RESPONSE ***"
160   WAIT 3000
170   GOTO Menu
180   PRINT PAGE
190   ON Ef GOTO 200,Chgset,Add,Addo,Delete,Merge,Quit
200  !
210   INPUT "DO YOU WANT SUBFILE ADDRESSING (0) OR ABSOLUTE ADDRESSING (1)?",S
220   IF (S=0) OR (S=1) THEN 260
230   BEEP
240   PRINT PAGE;"*** IMPROPER RESPONSE ***"
250   GOTO 210
260   PRINT PAGE
270   IF S=1 THEN Absol
280   INPUT "WHICH SUBFILE #?",Sn
290   IF (Sn>0) AND (Sn<=Nsub) AND (FRACT(Sn)=0) THEN 330
300   BEEP
310   PRINT PAGE;"*** SUBFILE NUMBER NOT DEFINED ***"
320   GOTO 280
330   PRINT PAGE
340   DISP "WHICH OBSERVATION IN SUBFILE ";Nsub$(Sn);
350   INPUT Subn
360   IF (Subn>0) AND (Subn<=Subf(Sn+1)-Subf(Sn)) AND (FRACT(Subn)=0) THEN 400
370   BEEP
380   PRINT PAGE;"*** SUBFILE OBSERVATION ADDRESS OUT OF RANGE ***"
390   GOTO 340
400   PRINT PAGE
410   INPUT "WHICH VARIABLE #",Vn
420   IF (Vn>0) AND (Vn<=Nv) AND (FRACT(Vn)=0) THEN 460
430   BEEP
440   PRINT PAGE;"*** IMPROPER VARIABLE NUMBER ***"
450   GOTO 410
460   PRINT PAGE
470   DISP "OLD VALUE = ";X(Subf(Sn)+Subn,Vn);"   NEW VALUE?";
480   INPUT "",A
490   PRINT USING 510;"OBS#:",Subn,"  IN SUBFILE:",Nsub$(Sn),"  OLD VALUE:",X(Su
bf(Sn)+Subn,Vn),"  NEW VALUE:",A
500   X(Subf(Sn)+Subn,Vn)=A
510   IMAGE 5A,4D,13A,5A,12A,5D.5D,12A,5D.5D
520   INPUT "ANOTHER VALUE?",S$
530   GOSUB Check
540   IF Check=1 THEN 520
550   IF UPC$(S$[1,1])="Y" THEN 340
560   INPUT "ANOTHER SUBFILE?",S$
570   GOSUB Check
580   IF Check=1 THEN 560
590   IF UPC$(S$[1,1])="Y" THEN 280
600   GOTO Menu
610 Absol: INPUT "WHICH ABSOLUTE OBSERVATION NUMBER?",Ob
620   IF (Ob>0) AND (Ob<=Nobs) AND (FRACT(Ob)=0) THEN 660
630   PRINT PAGE;"*** IMPROPER OBSERVATION NUMBER ***"
640   BEEP
650   GOTO Absol
660   PRINT PAGE
670   INPUT "WHICH VARIABLE NUMBER?",Vn
680   IF (Vn>0) AND (Vn<=Nv) AND (FRACT(Vn)=0) THEN 720
690   BEEP
700   PRINT PAGE;"*** IMPROPER VARIABLE NUMBER ***"
710   GOTO 670
720   PRINT PAGE
730   DISP "OLD VALUE ";X(Ob,Vn);"   NEW VALUE =  ";
```

```
740   INPUT A
750   PRINT USING 760;"OBS #:",Ob,"  OLD VALUE:",X(Ob,Vn),"  NEW VALUE:",A
760   IMAGE 6A,4D,12A,5D.5D,12A,5D.5D
770   X(Ob,Vn)=A
780   INPUT "CHANGE ANOTHER OBSERVATION?",S$
790   GOSUB Check
800   IF Check=1 THEN 780
810   IF UPC$(S$[1,1])="Y" THEN Absol
820   GOTO Menu
830 Chgset: ! PRINT PAGE
840   INPUT "WHICH SUBFILE #?",Sn
850   IF (Sn>0) AND (Sn<=Nsub) AND (FRACT(Sn)=0) THEN 890
860   BEEP
870   PRINT PAGE;"***SUBFILE NUMBER NOT DEFINED***"
880   GOTO 840
890   PRINT PAGE
900   PRINT "WHICH SET OF OBSERVATIONS IN SUBFILE ";Nsub$(Sn)
910   INPUT Subn
920   IF (Subn>0) AND (Subn<=Subf(Sn+1)-Subf(Sn)) AND  FRACT(Subn)=0) THEN 960
930   BEEP
940   PRINT PAGE;"***SUBFILE OBSERVATION ADDRESS OUT OF RANGE***"
950   GOTO 900
960   PRINT PAGE
970   PRINT "OLD VALUES FOR OBSERVATION ";Subn
980   PRINT LIN(2)
990   Jj=0
1000  FOR A=1 TO Nv
1010    Jj=Jj+1
1020    IF Jj<=5 THEN GOTO 1050
1030    Jj=1
1040    PRINT USING 1060
1050    PRINT USING 1070;X(Subf(Sn)+Subn,A)
1060    IMAGE /
1070    IMAGE #,5X,5D.4D
1080  NEXT A
1081  PRINT LIN(5)
1090  PRINT "ENTER NEW VALUES FOR OBSERVATION ";Subn
1100  PRINT LIN(2)
1110  DIM Aa(20)
1120  MAT INPUT Aa
1130  FOR A=1 TO Nv
1140    X(Subf(Sn)+Subn,A)=Aa(A)
1150  NEXT A
1160  INPUT "ANOTHER SET OF OBSERVATIONS?",S$
1170  GOSUB Check
1180  IF Check=1 THEN 1160
1190  IF UPC$(S$[1,1])="Y" THEN 900
1200  GOTO Menu
1210 Add:! PRINT PAGE
1220  IF Nsub+1<=20 THEN 1270
1230  BEEP
1240  DISP "*** MAXIMUM NUMBER OF SUBFILES EXCEEDED ***"
1250  WAIT 3000
1260  GOTO Menu
1270  INPUT "NUMBER OF OBSERVATIONS IN SUBFILE?",No
1280  IF Nobs+No<=500 THEN 1320
1290  PRINT PAGE;"*** MAXIMUM NUMBER OF OBSERVATIONS EXCEEDED ***"
1300  BEEP
1310  GOTO 1270
1320  PRINT PAGE
1330  Nobs=Nobs+No
1340  Nsub=Nsub+1
1350  INPUT "SUBFILE NAME? (5 CHARACTERS OR LESS)",Nsub$(Nsub)
1360  Subf(Nsub+1)=Nobs
1370  FOR A=1 TO No
1380     FOR B=1 TO Nv
1390        DISP "OBERVATION # ";A;"  VARIABLE # ";B;" IS ";
1400        INPUT X(Nobs-No+A,B)
```

```
1410      NEXT B
1420   NEXT A
1430   INPUT "ANOTHER SUBFILE?",S$
1440   GOSUB Check
1450   IF Check=1 THEN 1430
1460   IF UPC$(S$[1,1])="Y" THEN 1020
1470   GOTO Menu
1480 Addo:! INPUT "DO YOU WANT TO ADD AN OBSERVATION?",S$
1490   IF Nobs+1<=500 THEN 1540
1500   BEEP
1510   DISP "*** MAXIMUM NUMBER OF OBSERVATIONS EXCEEDED ***"
1520   WAIT 3000
1530   GOTO Menu
1540   INPUT "DO YOU WANT SUBFILE ADDRESSING (0) OR ABSOLUTE ADDRESSING (1)?",S
1550   IF (S=0) OR (S=1) THEN 1590
1560   BEEP
1570   PRINT PAGE;"*** IMPROPER RESPONSE ***"
1580   GOTO 1540
1590   PRINT PAGE
1600   IF S=0 THEN Suba
1610   INPUT "WHICH ABSOLUTE OBSERVATION NUMBER?",N
1620   IF (N>0) AND (N<=Nobs) AND (FRACT(N)=0) THEN 1660
1630   BEEP
1640   PRINT PAGE;"*** IMPROPER OBSERVATION NUMBER ***"
1650   GOTO 1610
1660   PRINT PAGE
1670   FOR A=1 TO Nv
1680      FOR B=Nobs TO N STEP -1
1690         X(B+1,A)=X(B,A)
1700      NEXT B
1710   NEXT A
1720   FOR A=1 TO Nv
1730      DISP "INPUT VARIABLE #";A;" FOR OBSERVATION #";N;
1740      INPUT "",X(N,A)
1750   NEXT A
1760   FOR A=1 TO Nsub+1
1770      IF N<=Subf(A) THEN Subf(A)=Subf(A)+1
1780   NEXT A
1790   Nobs=Nobs+1
1800   INPUT "ANOTHER OBSERVATION?",S$
1810   GOSUB Check
1820   IF Check=1 THEN 1800
1830   IF UPC$(S$[1,1])="Y" THEN 1610
1840   GOTO Menu
1850 Suba: INPUT "WHAT SUBFILE # DO YOU WANT TO ADD AN OBSERVATION TO?",Sun
1860   IF (Sun>0) AND (Sun<=Nsub) AND (FRACT(Sun)=0) THEN 1900
1870   PRINT PAGE;"*** IMPROPER SUBFILE NUMBER ***"
1880   BEEP
1890   GOTO Suba
1900   PRINT PAGE
1910   INPUT "WHAT OBSERVATION NUMBER DO YOU WANT TO ADD?",Obnum
1920   IF Nobs+1<=500 THEN 1960
1930   BEEP
1940   PRINT PAGE;"*** MAXIMUM NUMBER OF OBSERVATIONS EXCEEDED ***"
1950   GOTO Menu
1960   PRINT PAGE
1970   IF (Obnum>0) AND (Obnum<=Subf(Sun+1)-Subf(Sun)+1) AND (FRACT(Obnum)=0) THE
N 2010
1980   BEEP
1990   PRINT PAGE;"*** IMPROPER OBSERVATION NUMBER FOR THIS SUBFILE ***"
2000   GOTO 1910
2010   PRINT PAGE
2020   FOR A=Nobs TO Subf(Sun)+Obnum STEP -1
2030      FOR B=1 TO Nv
2040         X(A+1,B)=X(A,B)
2050      NEXT B
2060   NEXT A
2070   FOR A=1 TO Nv
```

```
2080     DISP "SUBFILE ";Nsub$(Sun);"  OBSERVATION ";Obnum;" VARIABLE #";A;" IS"
;
2090     INPUT X(Obnum+Subf(Sun),A)
2100  NEXT A
2110  Nobs=Nobs+1
2120  FOR A=Sun+1 TO Nsub+1
2130     Subf(A)=Subf(A)+1
2140  NEXT A
2150  INPUT "ANOTHER ADDITION TO THIS SUBFILE?",S$
2160  GOSUB Check
2170  IF Check=1 THEN 2150
2180  IF UPC$(S$[1,1])="N" THEN 2200
2190  GOTO 1910
2200  INPUT "ANOTHER ADDITION TO ANOTHER SUBFILE?",S$
2210  GOSUB Check
2220  IF Check=1 THEN 2200
2230  IF UPC$(S$[1,1])="Y" THEN Suba
2240  GOTO Menu
2250 Delete:! INPUT "DO YOU WANT TO DELETE AN OBSERVATION?",S$
2260  INPUT "DO YOU WANT ABSOLUTE ADDRESSING (0) OR SUBFILE ADDRESSING (1)?",S
2270  IF (S=0) OR (S=1) THEN 2310
2280  BEEP
2290  PRINT PAGE;"*** IMPROPER RESPONSE ***"
2300  GOTO Delete
2310  PRINT PAGE
2320  IF S=1 THEN Subdel
2330  INPUT "ABSOLUTE OBSERVATION NUMBER?",On
2340  IF (On>0) AND (On<=Nobs) AND (FRACT(On)=0) THEN 2380
2350  PRINT PAGE;"*** IMPROPER OBSERVATION NUMBER ***"
2360  BEEP
2370  GOTO 2330
2380  PRINT PAGE;"      OBSERVATION NUMBER ";On
2390  FOR A=1 TO Nv
2400    PRINT "VARIABLE #";A;" = ";X(On,A)
2410  NEXT A
2420  INPUT "DELETE?",S$
2430  GOSUB Check
2440  IF Check=1 THEN 2420
2450  IF UPC$(S$[1,1])="N" THEN Abort
2460  FOR A=On TO Nobs
2470     FOR B=1 TO Nv
2480        X(A,B)=X(A+1,B)
2490     NEXT B
2500  NEXT A
2510  FOR A=1 TO Nsub+1
2520     IF Subf(A)>On THEN Subf(A)=Subf(A)-1
2530  NEXT A
2540  Nobs=Nobs-1
2550 Abort: INPUT "DELETE ANOTHER OBSERVATION?",S$
2560  GOSUB Check
2570  IF Check=1 THEN Abort
2580  IF UPC$(S$[1,1])="Y" THEN 2330
2590  GOTO Menu
2600 Subdel: INPUT "SUBFILE NUMBER OF DELETED OBSERVATION?",Sn
2610  IF (Sn>0) AND (Sn<=Nsub) AND (FRACT(Sn)=0) THEN 2650
2620  BEEP
2630  PRINT PAGE;"*** IMPPOPER SUBFILE NUMBER ***"
2640  GOTO Subdel
2650  PRINT PAGE
2660  INPUT "OBSERVATION NUMBER IN SUBFILE?",On
2670  IF (On>0) AND (On<=Subf(Sn+1)-Subf(Sn)) AND (FRACT(On)=0) THEN 2710
2680  BEEP
2690  PRINT PAGE;"*** IMPROPER OBSERVATION NUMBER FOR THIS SUBFILE ***"
2700  GOTO 2660
2710  PRINT PAGE;"SUBFILE: ";Nsub$(Sn);"    OBSERVATION # ";On
2720  FOR A=1 TO Nv
2730     PRINT "VARIABLE # ";A;" IS ";X(On+Subf(Sn),A)
2740  NEXT A
```

```
2750  INPUT "DELETE?",S$
2760  GOSUB Check
2770  IF Check=1 THEN 2750
2780  IF UPC$(S$[1,1])="N" THEN Obor
2790  FOR A=Subf(Sn)+On TO Nobs
2800     FOR B=1 TO Nv
2810        X(A,B)=X(A+1,B)
2820     NEXT B
2830  NEXT A
2840  FOR A=1 TO Nsub+1
2850     IF Subf(A)>Subf(Sn)+On THEN Subf(A)=Subf(A)-1
2860  NEXT A
2870  Nobs=Nobs-1
2880 Obor: INPUT "ANOTHER DELETION FROM THIS SUBFILE?",S$
2890  GOSUB Check
2900  IF Check=1 THEN Obor
2910  IF UPC$(S$[1,1])="Y" THEN 2660
2920  INPUT "ANOTHER DELETION FROM ANOTHER SUBFILE?",S$
2930  GOSUB Check
2940  IF Check=1 THEN 2920
2950  IF UPC$(S$[1,1])="Y" THEN Subdel
2960  GOTO Menu
2970 Merge:  INPUT "FILE NAME?",Filn$
2980  ON ERROR GOTO Undefined
2990  ASSIGN #1 TO Filn$
3000  READ #1;Nv2,Nobs2,Nsub2
3010  IF Nobs2+Nobs<=500 THEN 3060
3020  BEEP
3030  DISP "*** MAXIMUM NUMBER OF OBSERVATIONS EXCEEDED ***"
3040  WAIT 2000
3050  GOTO Menu
3060  IF Nsub2+Nsub<=20 THEN 3110
3070  BEEP
3080  DISP "*** MAXIMUM NUMBER OF SUBFILES EXCEEDED ***"
3090  WAIT 2000
3100  GOTO Menu
3110  IF Nv2=Nv THEN 3160
3120  BEEP
3130  DISP "*** VARIABLE NUMBER MISMATCH BETWEEN MERGED FILES ***"
3140  WAIT 2000
3150  GOTO Menu
3160  FOR A=1 TO Nsub2
3170     READ #1;Nsub$(A+Nsub)
3180  NEXT A
3190  READ #1;A
3200  FOR A=2 TO Nsub2+1
3210     READ #1;Subf(A+Nsub)
3220     Subf(A+Nsub)=Subf(A+Nsub)+Subf(Nsub+1)
3230  NEXT A
3240  FOR A=1 TO Nobs2
3250     FOR B=1 TO 20
3260        READ #1;X(A+Nobs,B)
3270     NEXT B
3280  NEXT A
3290  Nsub=Nsub+Nsub2
3300  Nobs=Nobs+Nobs2
3310  OFF ERROR
3320  GOTO Menu
3330 Undefined: IF ERRN=56 THEN 3380
3340  BEEP
3350  DISP ERRM$
3360  PAUSE
3370  GOTO Menu
3380  BEEP
3390  PRINT PAGE;"*** FILENAME IS UNDEFINED ***"
3400  GOTO Merge
3410 Quit: !
3420  PRINT PAGE
```

```
3430  DISP "RETURNING TO MAIN PROGRAM"
3440  LOAD "AUTOST",Restart
3450  !
3460  ! CHECKS YES AND NO RESPONSES
3470 Check: Check=0
3480  PRINT PAGE
3490  IF (UPC$(S$[1,1])="Y") OR (UPC$(S$[1,1])="N") THEN 3530
3500  BEEP
3510  PRINT PAGE;"*** IMPROPER RESPONSE - TRY AGAIN ***"
3520  Check=1
3530  RETURN
```

APPENDIX B: HEWLETT-PACKARD USERS MANUALS

Advanced Programming ROM Manual. Hewlett Packard Co., Part No. 09845-92065, Feb. 1980, 108 p.

BASIC Language Interfacing Concepts. Hewlett Packard Co., Part No. 09835-90600, Sept. 1979, 189 p.

Graphics ROM Manual. Hewlett Packard Co., Part No. 09845-91050, May 1979, 203 p.

I/O ROM Manual. Hewlett Packard Co., Part No. 09845-92060, Aug. 1980, 192 p.

Mass Storage ROM Manual. Hewlett Packard Co., Part No. 09845-92070, Feb. 1980, 132 p.

System 45 Operating and Programming Manual. Hewlett Packard Co., Part No. 09845-92000, Feb. 1980, 302 p.